Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Mathematical Sciences

Johannes Padel

Predicting Dry Bulk Vessel Destinations Using Historical AIS Data

Master's thesis in Applied Physics and Mathematics
Supervisor: Jo Eidsvik
Co-supervisor: Kristian Amundsen Ruud

July 2023

NTNU
Norwegian University of
Science and Technology

Johannes Padel

Predicting Dry Bulk Vessel Destinations Using Historical AIS Data

Master’s thesis in Applied Physics and Mathematics
Supervisor: Jo Eidsvik
Co-supervisor: Kristian Amundsen Ruud
July 2023

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Mathematical Sciences

NTNU
Norwegian University of
Science and Technology

Abstract

Global trade and economic growth greatly rely on the shipping industry, with a key role played by dry bulk shipping. This form of shipping deals with transporting loose bulk cargo, such as grains iron, and coal, and is crucial for various sectors. However, the industry encounters difficulties stemming from the volatile market conditions, which make it challenging to anticipate vessel destinations accurately. As a result, scheduling, routing, and resource allocation suffer from inefficiencies. To tackle these challenges, the application of machine learning can significantly enhance operational efficiency and decision-making in the shipping industry by more accurately predicting vessel destinations.

This thesis investigates the application of machine learning techniques to predict the destinations of dry bulk vessels in the shipping industry. The study focuses on the multiclass classification problem of predicting port-to-port, laden, and ballast voyages. A comprehensive pre-processing of the Automatic Identification System signals was performed to create trajectories. Trajectory similarity measures were used to collect an initial prediction, which was used as a feature in the machine learning model, along with other relevant information. The XGBoost algorithm was employed for the classification task, with separate models created for the largest sub-segments of dry bulk vessels, namely Very Large Ore Carriers, Capesize, Panamax, and Supramax. The performance of the models was evaluated using various metrics, including port accuracy, port frequency-based decision accuracy, cluster accuracy, country accuracy, Average Prediction Distance Error, Median Prediction Distance Error, and the Brier score.

The overall results varied between 50-83 % for port prediction accuracy, and between 75-96 % for country accuracy. The models performed best on larger vessels, particularly the Very Large Ore Carriers for port-to-port predictions, and achieved the best overall result of 83 % accuracy for Capesize ballast voyages. The laden voyages where the hardest to predict, with results varying between 24-64 %. Permutation feature importance and SHAP values were used to investigate feature importance, revealing a correlation between real-world events and the model's predictions. This thesis contributes to the understanding of how machine learning can be used to improve operational efficiency in the shipping industry.

Sammendrag

Den globale handelen og økonomiske veksten er sterkt avhengig av shippingindustrien, der tørrbulkfrakt spiller en nøkkelrolle. Denne formen for frakt håndterer transport av løst bulkgods, som korn, jern og kull, og er avgjørende for ulike sektorer. Imidlertid står bransjen overfor utfordringer som skyldes volatiliteten i markedet, noe som gjør det utfordrende å nøyaktig predikere destinasjonene for skipene. Som et resultat lider planlegging, rutevalg og ressursallokering av diverse ineffektiviteter i markedet. For å håndtere disse utfordringene kan anvendelsen av maskinlæring betydelig forbedre driftseffektiviteten og beslutningstakingen i skipsfartsindustrien ved å mer nøyaktig predikere skipenes destinasjoner.

Denne oppgaven undersøker bruken av maskinlæringsteknikker for å forutsi destinasjonene til tørrbulkskip i shippingindustrien. Studien fokuserer på flerklassifiseringsproblemet med å forutsi havn-til-havn, lastede og ballastreiser. En omfattende preprossesring av Automatic Identification System (AIS) signaler ble utført for å lage skipsbaner. Matematiske distanse modeller for baner ble brukt til å lage en innledende prediksjon, som ble brukt som variabel i maskinlæringsmodellen, sammen med annen relevant informasjon. XGBoost-algoritmen ble brukt for flerklassifiseringsoppgaven, med separate modeller opprettet for de største sub-segmentene av tørrbulkskip, nemlig Very Large Ore Carriers, Capesize, Panamax og Supramax. Modellenes ytelse ble evaluert ved hjelp av forskjellige metrikker, inkludert havnenøyaktighet, beslutningsnøyaktighet basert på havnefrekvens, klyngenøyaktighet, landsnøyaktighet, gjennomsnittlig feil i prediksjonsavstand, median feil i prediksjonsavstand, og Brier-scoren.

Resultatene varierte betydelig, der nøyaktigheten for havne-prediksjon varierte mellom 50-82 %, og for landsnøyaktighet mellom 75-96%. Modellene presterte best på større skip, spesielt Very Large Ore Carriers for havn-til-havn-prediksjoner, og oppnådde det beste samlede resultatet med 82 % nøyaktighet for Capesize ballastreiser. De lastede reisene var de vanskeligste å forutsi, med resultater som varierte mellom 24-64 %. Permutasjonsvariabelbetydning og SHAP-verdier ble brukt til å undersøke variabelbetydning, og avslørte en korrelasjon mellom virkelige hendelser og modellens prediksjoner. Denne oppgaven bidrar til forståelsen av hvordan maskinlæring kan brukes til å forbedre driftseffektiviteten i shippingindustrien.

Preface

This theis represents the culmination of my five-year Master of Science program in Physics and Mathematics at NTNU. My chosen specialization is Industrial Mathematics, where I have further specialized in the field of statistics.

The topic of this thesis was chosen due to my interest in applying mathematics and statistics to real-world scenarios. This work exemplifies how powerful theoretical concepts can be when applied to practical issues, particularly those that have substantial implications on global activities, such as shipping.

I would like to express my profound gratitude to Professor Jo Eidsvik from the Institute of Mathematics at NTNU. His academic insights, guidance, and weekly support were pivotal in shaping this thesis. My thanks also extend to Kristian Amundsen Ruud from Astrup Fearnley CODE. His contribution was invaluable in terms of providing the problem statement and the essential data that formed the backbone of this research. Without his practical input resource provision and expert knowledge in the shipping industry, this thesis would not have been possible.

As I present this work, I am deeply grateful to everyone who contributed to my journey, especially my partner, family and classmates, and I am excited about the opportunities that the future holds.

Trondheim July 2023 Johannes Padel

Contents

Chapter 1

Introduction

1.1 The shipping market and dry bulk vessels

The shipping industry is of paramount importance for the global economy, as it is responsible for transporting a significant portion of international trade. According to the United Nations Conference on Trade and Development (UNCTAD), sea transport accounted for approximately 80% of the world's goods by volume and 70% of goods by value in 2018 (Sirimanne et al. 2019).

The shipping trade process involves various actors, including shippers, receivers, shipowners, charterers, shipbrokers, and traders, each playing distinct roles (Stopford 2008). Shippers are typically producers or traders who have goods to transport, and receivers are typically the consumers or buyers of these goods. The shipping process begins when a shipper needs to transport goods from one location to another. A charterer is a person or company who hires a ship from a shipowner for a particular voyage (voyage charter) or for a particular period of time (time charter). They negotiate chartering contracts directly with the shipowner or through a shipbroker. The charterer is responsible for providing the cargo and deciding the ports of call. Shipowners, who own the vessels, provide the service of transporting goods. They can operate their own ships or lease them out to charterers. Shipbrokers act as intermediaries between shipowners and charterers. They negotiate the terms of the charter on behalf of both parties, ensuring that all the necessary contractual obligations are understood and agreed upon.

The sequence of events in the shipping trade process typically goes as follows: A trader or producer (shipper) who has goods to ship contacts a shipbroker or charterer to find a suitable ship. This is not always the case, an example being the Valemax fleet, where the mining company Vale S.A, being the shipper, owns their own fleet (Papadionysiou 2014). However, typically a shipbroker is required. Thus it is the shipper that decides the final destination of the vessels, due to the nature of the trade. The shipbroker then negotiates with shipowners to find a vessel that meets the shipper's needs in terms of size, type, and availability. Once the shipbroker finds a suitable ship, they negotiate the terms of the charter contract, including freight rate, duration, and route. The ship then transports the goods to the specified destination, and upon arrival, the cargo is unloaded by the receiver. The shipowner is paid the agreed freight rate, out of which the shipbroker receives a commission.

Dry bulk vessels are a type of shipping vessel that are designed to transport large quantities of dry cargo, such as coal, iron ore, grain, and other commodities (Stopford 2008). They are distinct from other types of vessels due to their size, cargo capacity, and operational requirements. Dry bulk vessels are equipped with large, open holds that are used to store the dry cargo, which are clearly seen in Figure 1.1. These holds are typically box-like in shape and are covered by large hatches on the deck of the ship to protect the cargo from the elements. The cargo is loaded onto the ship through these hatches, often using a crane or conveyor belt system. For instance, large cranes are typically used for solid commodities such as iron ore and coal, while specialized equipment like grain elevators or pneumatic conveyors are used for loading agricultural products like grain (Kendall 2012).

The dry bulk vessel industry is marked by significant volatility (Alizadeh and Nomikos 2013). A key contributor to this instability is the imbalance between supply and de-

Figure 1.1: An example of a dry bulk vessel

mand, made worse by the decentralized nature of the spot market and a deficiency of accurate, real-time information about cargo and vessel availability (Jugović et al. 2015).

Astrup Fearnley Code (AF Code) has supplied a graph displaying the volatility of the spot market. Figure 1.2 displays the spot market indices for Capesize, Panamax, and Supramax sub-segments of dry bulk vessels from 2019 to 2022, for time charter contracts(TC). TC is a type of contract for the hire of a ship, where the shipowner provides the vessel, crew, insurance, and other necessary provisions, but the charterer controls the voyages and pays for the fuel and port charges (Agnolucci et al. 2014). The spot market index is a measure of the current rate for shipping a particular type of cargo on a particular type of vessel, quoted in U.S. dollars per day (USD/day). The rates are determined by the supply and demand of vessels and cargoes in the market at any given moment.

In the current system, the bid for bulk cargo contracts is heavily dependent on how close a ship is to the loading port and whether it can reach there within the required timeframe. However, lack of information about other ships potentially going to the

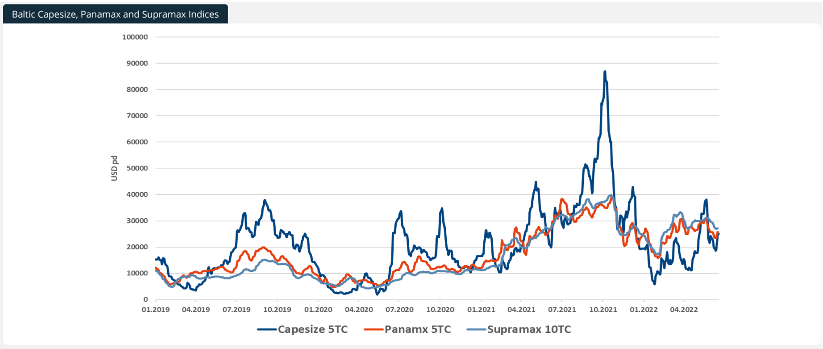


Figure 1.2: Spot market indices between 2019 and 2022 for different sub-segments of dry bulk vessels.

same port introduces uncertainty. This unpredictability in the supply of ships, both in terms of location and timing, can lead to inefficiencies in the market and significant economic challenges (Jing et al. 2008).

1.2 AIS signals and trajectories

The Automatic Identification System (AIS), established by the International Maritime Organisation (IMO) under the Safety Of Life At Sea (SOLAS) convention in the early 2000s, was designed to standardize and enhance maritime safety (Joseph and Dalaklis 2021). The convention's main aim is to outline basic safety standards that all vessels from participating countries must adhere to. The SOLAS convention prioritizes the security and safe navigation of ships, which led to the creation of the AIS. The AIS contributes to this goal by providing real-time location updates to both other ships and onshore stations within their radio range, thus reducing the likelihood of ship collisions.

AIS signals are transmitted using a specific format that includes a set of binary data packets. The data packets contain information about the vessel, such as its unique identifier, GPS position, course, speed, and other relevant information. A typical AIS data packet can have the following format:

- **Message Type**: Identifies the type of message
- **Repeat Indicator**: Number of times a message has been transmitted
- **IMO nr**: Unique identifier for the vessel
- **Navigation Status**: Specifies the vessel's navigation status
- **Latitude**: Latitude of the vessel's location
- **Longitude**: Longitude of the vessel's location
- **Speed Over Ground**: Speed of the vessel over the ground
- **Destination**: Destination, manually inputted
- **Course Over Ground**: Course of the vessel over the ground

The destination field is of interest here because it is the focus of this thesis work. According to Abdallah et al. (2019), only around 38 % of AIS signals contain accurate destinations. This can be due to geopolitical or economic reasons, for example that the trader of goods does not want rivals to know which ports the vessels are traveling to, and by doing so one gains an economic advantage.

Table 1.1: Typical example of the manually inputted destination field in the AIS signals

Date	Departure Port	Arrival Port	**Destination**
06.05.23	NEWCASTLE	RIZHAO	ZHOUSHAN
15.05.23	NEWCASTLE	RIZHAO	ZHOUSHAN
22.05.23	NEWCASTLE	RIZHAO	RIZHAO

In Table 1.1, we observe a particular vessel taking a route from NEWCASTLE AUSTRALIA to RIZHAO, at three different time stages of the journey, however the field "Destination" changes after two weeks, perhaps because of an instruction from the trader.

Since AIS signals are available and provide information about vessel trajectories, they can be utilized to create trajectories that help in predicting vessel destinations. By analyzing historical AIS data and considering factors such as vessel behavior, route patterns, and destination patterns of similar vessels, it becomes possible to make informed predictions about the future destinations of vessels.

1.3 Problem statement

The key problem to be addressed in this thesis is the prediction of dry bulk vessel destinations using multiclass classification models. The suggested models utilize historical AIS data and account for the probabilistic nature of other important features.

The goal is to build and compare trajectories using AIS data, providing a spatial solution to the problem. This spatial perspective is then integrated with other static information for machine learning applications, considering the probabilistic aspects of other important features. The approach is not only to predict port-to-port voyages, but also laden and ballast voyages, meaning from the loading of cargo to the unloading and vice versa. Given that the decision of the vessels' destinations is made by the trader of goods, the models will incorporate features related to these decision-makers.

Addressing this problem could significantly reduce the inherent volatility of the dry bulk vessel industry, contributing to improved market stability and economic efficiency. The challenge lies in the selection of appropriate machine learning models, such as XGBoost, and features that can effectively deal with the complexity and stochasticity of the problem.

Similar studies have been conducted for prediction of destination ports, such as in Roşca et al. (2018), Zhang et al. (2020) and Omholt-Jensen (2021), however the main contribution for this thesis lies in the prediction of laden and ballast voyages, which has not, to the best of our knowledge, been previously explored in depth for dry-bulk vessels.

1.4 Outline

In chapter 2, we investigate the dataset derived from AIS signals. We introduce subsegments in the dry bulk vessel segment (Section 2.1), explain how AIS Trade Flow systems work (Section 2.2), provide definitions for different types of vessel voyages (Section 2.3), and explain how we create trajectories (Section 2.4). We introduce Port Clustering(Section 2.5), then perform an exploratory data analysis on the full dataset (Section 2.6).

Chapter 3 is dedicated to explaining the research methodology. We discuss the multiclass classification problem notation (Section 3.1), trajectory-based features (Sections 3.2). Afterwards, we discuss how we prepare the machine learning datasets (Sections 3.3), how we handle encoding of categorical features (Sections 3.4), and present our machine learning models (Section 3.5). Finally, we explain our evaluation metrics (Sections 3.6) and the concepts of feature importance (Sections 3.7).

Chapter 4 analyzes and discusses the results obtained from our machine learning models for the three different voyage types: port-to-port voyages (Sections 4.1), laden voyages (Sections 4.2), and ballast voyages (Sections 4.3). We discuss the results in the context of machine learning performances and feature importance, and tie this up to the dry bulk trade market.

The final chapter summarizes the findings of our research, highlighting the implications and potential future directions for the shipping industry.

Chapter 2

AIS Dataset

2.1 Sub-segments in the dry bulk vessel segment

The dry bulk vessel segment encompasses several sub-segments that cater to different transportation needs in the bulk cargo industry. In this thesis, the focus is on four main sub-segments: Very Large Ore Carriers (VLOC), Capesize, Panamax, and Supramax (Stopford 2008). Figure 2.1 shows examples of these four sub-segments, along with their deadweight. A more detailed explanation follows below:

VLOC: VLOCs are specialized bulk carriers designed to transport large quantities of iron ore and other minerals. They are among the largest bulk carriers, typically ranging from 200,000 to 400,000 deadweight tons (DWT). VLOCs are primarily utilized in long-haul routes, often transporting iron ore from major exporting countries like Brazil and Australia to industrial centers such as China. Due to their size, VLOCs require deep-water ports for loading and unloading.

Capesize: Capesize vessels derive their name from their need to navigate the Cape of

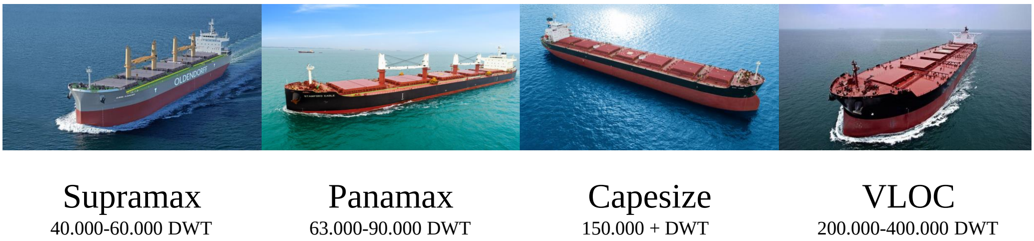


Figure 2.1: Images displaying the four different dry-bulk sub-segments this thesis focuses on, along with their typical deadweight.

Good Hope and Cape Horn. These vessels are too large to pass through the Panama Canal or Suez Canal and therefore must circumnavigate the southern tips of Africa or South America. They are the largest bulk carriers(except VLOC), usually exceeding 150,000 DWT. Capesize vessels are predominantly used to transport commodities like iron ore and coal on intercontinental routes, such as from Brazil or Australia to China or Europe.

Panamax: Panamax vessels are named after their size limitations to fit the maximum dimensions of the Panama Canal locks. These vessels typically have a DWT ranging from 63.000 to 90.000 tons. Panamax vessels are versatile and can transport various dry bulk commodities, including grains, coal, and ores. They are commonly employed on medium-range routes, such as shipments between the Americas, Europe, and Asia. The Panama Canal expansion in 2016 has increased the maximum size of vessels that can pass through, known as Neopanamax, enabling larger vessels in this segment.

Supramax: Supramax vessels are smaller-sized bulk carriers, typically ranging from 40.000 to 60.000 DWT. They are highly versatile and well-suited for accessing ports with restrictions on draft and infrastructure limitations. Supramax vessels are commonly employed for transporting various dry bulk cargoes, including grains, coal, minerals, and steel products. They provide flexibility and are able to serve a wide range of routes, including both short-haul and long-haul voyages.

2.2 AIS Trade Flow systems

AIS Trade Flow systems work by leveraging the data transmitted by vessels globally to provide real-time and historical insights into maritime trade activities (Halden 2019). The goal is to build a system for defining trade between ports using the AIS signals. Defining port areas in AIS Trade Flow systems is a complex process that involves both geographical and operational considerations. Geographically, a port area is typically defined by a set of coordinates that outline the physical boundaries of the port and its surrounding waters. This can include berths, anchorages, and sometimes even the approach channels. Determining whether a ship is inside a port area often involves comparing the ship's current AIS-reported coordinates with the defined boundaries of the port. If the ship's position falls within these boundaries, it is considered to be inside the port area. However, it is not just about geographical location. For example, a ship might be within the geographical boundaries of a port, but if it is just passing through without stopping or engaging in port activities, it might not be considered as being 'in port' from an operational perspective. Figure 2.2 represents an example of a vessel anchored outside of the Port of Singapore, however it is not close to where goods are transferred from vessel to shore(the red area), highlighting the complexity of deciding if a voyage is laden, ballast or just port-to-port.

To handle these complexities, AIS Trade Flow systems often use sophisticated algorithms to accurately determine port boundaries and to classify vessel behavior. This can include factors like the ship's speed, course, and historical patterns of behavior. By combining these different data points, these systems can provide a highly accurate picture of port activities and vessel movements. AF Code has built a system for this using the general principles, which has created a dataset containing the AIS signals along with portstops, loading and unloading. The destination field is also pre-processed and cleaned, which is explained in more detail later.

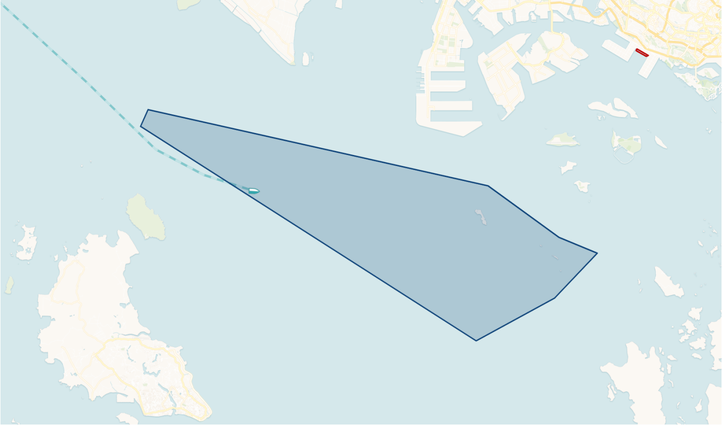


Figure 2.2: AIS Trade Flow example of a dry bulk vessel anchored outside of Singapore. The blue polygon represents an anchorage area, and the red area represents an area where the vessel has the capacity for loading and unloading.

2.3 Vessel voyage definitions

In this section, we introduce three definitions of vessel voyages, namely port-to-port voyages, laden voyages, and ballast voyages. Each of these types of voyages poses distinct challenges, and predicting them involves addressing three separate problems.

2.3.1 Port-to-port voyage definition

A port-to-port voyage refers to the journey of a vessel from the departure port to the destination port. This definition is commonly used for the analysis of maritime transportation networks, as it simplifies the representation of vessel movements between ports. The port-to-port definition can be formally described as follows:

$$V_{ij} = \{(P_i, P_j) : P_i \in \mathcal{P}, P_j \in \mathcal{P}, P_i \neq P_j\}, \tag{2.1}$$

where V_{ij} represents the voyage from port i to port j, $\mathcal{P}$ is the set of all ports in the network of available ports in the dataset, and P_i and P_j are the departure and destination ports, respectively.

2.3.2 Laden voyage definition

The laden voyage definition concentrates on the movement of cargo, rather than the movement of the vessel itself. This definition is particularly relevant when assessing the performance of logistics chains and cargo flows in the context of maritime transportation. A laden voyage is defined as the journey from the point of cargo loading at the origin port to the point of cargo unloading at the destination port. This definition can be formally described as:

$$C_{ij} = \{(L_i, D_j) : L_i \in \mathcal{L}, D_j \in \mathcal{D}, L_i \neq D_j, \mathcal{D}, \mathcal{L} \subset \mathcal{P}\}, \tag{2.2}$$

where C_{ij} represents the cargo flow on a vessel between loading point L_i and unloading point L_j, $\mathcal{L}$ is the set of all cargo loading points, and $\mathcal{D}$ is the set of all cargo unloading points. Note that all unloading ports are a subset of all ports, such that $\mathcal{D} \subset \mathcal{P}$

2.3.3 Ballast voyage definition

The ballast voyage definition focuses on the movement of the vessel when it is not carrying cargo, rather than the movement of the vessel with cargo. A ballast voyage is defined as the journey from the port of cargo unloading as the origin port, to the port of cargo loading as the destination port. This definition can be formally described as:

$$B_{ij} = \{(D_i, L_j) : D_i \in \mathcal{D}, L_j \in \mathcal{L}, D_i \neq L_j, \mathcal{D}, \mathcal{L} \subset \mathcal{P}\}, \tag{2.3}$$

where B_{ij} represents the flow of the vessel in ballast condition between unloading point D_i and loading point D_j, $\mathcal{D}$ is the set of all cargo unloading points, and $\mathcal{L}$ is the set of all cargo loading points. Note that all loading ports are a subset of all ports.

These two definitions regarding unloading and loading are of most interest, since many vessels stop at ports for bunkering. An example is displayed in Figure 2.3, where a vessel is loaded in Port Louis, arrives in Singapore for refueling, bunkering or crew change, and travels on to Rizhao for unloading of cargo, and then finally travels ballast to Port Walcott for a new contract. The laden voyage in this example is between Port Louis and Rizhao, the ballast voyage between Rizhao and Port Walcott, and the voyages between each port is a port-to-port voyage. Prediction of the vessels movement between individual ports V_{ij}, is not always the the most valuable information, and has been done in previous studies (Yin et al. 2022). AF Code are mostly interested in modeling the flow of goods, therefore models predicting P_j, D_j and L_j, will be used.

2.4 Data and trajectory creation

As mentioned earlier, we need to create trajectories between ports using the AIS data. This section details this process, along with illustrations. The raw data received from AF Code corresponds to one AIS signal, with features such as IMO, coordinates and timestamp. It also includes information such as the departure port, expected arrival port (which is not known in a real-time scenario), sub-segment, and other important features relating to the vessel. This data contains more than a quarter billion rows, with more parts belonging to the smaller sub-segments. The precise amount is 258,510,150, and the distribution for the various sub-segments is displayed in Table 2.1. Many of these points contain errors or missing values that will be handled by simply discarding rows, due to the vast data amount available. A sample of 150,000

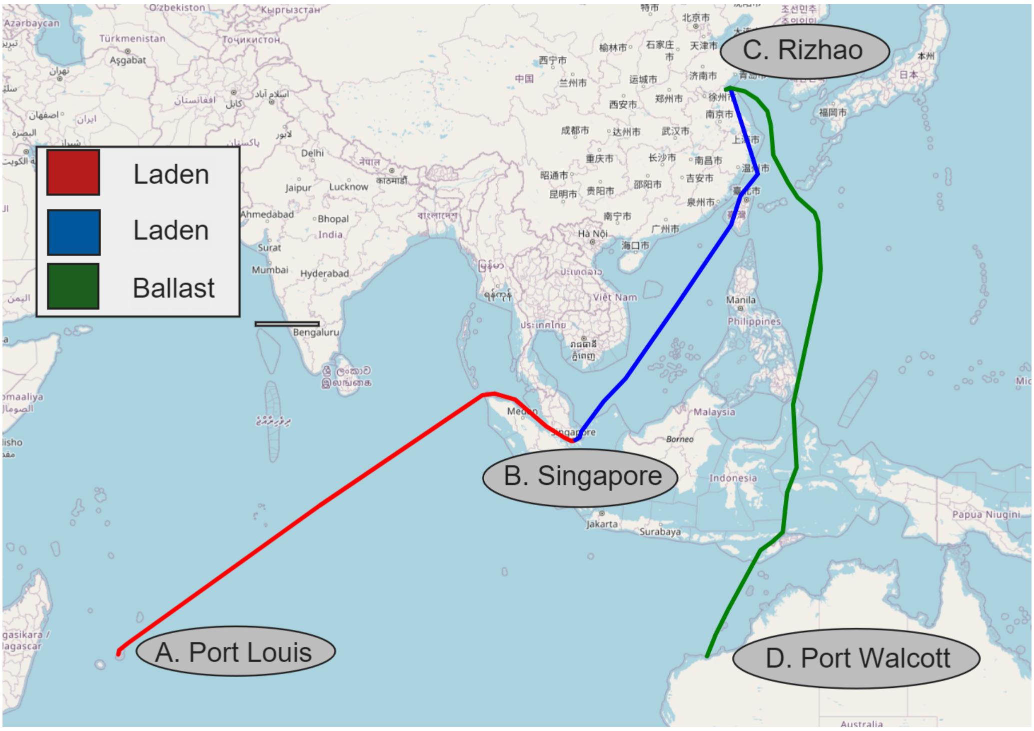


Figure 2.3: Trajectory of vessel loaded in Port Louis, bunkering in Singapore, unloaded in Rizhao, and loaded again in Port Walcott.

points is displayed in Figure 2.4, color-coded on sub-segment. This displays only 0.06% of the signals, showing the sheer scale of the data. In this figure one can clearly also see the common routes of the different sub-segments, that Capesize and VLOC travel around South Africa, while e.g Supramax tends to stay closer to land and travels through the Suez Canal. Panamax and Supramax are also more evenly spread out across the world compared to VLOC and Capesize vessels.

Vessel Type	Value
VLOC	8,925,113
Supramax	98,711,76
Panamax	95,005,262
Capesize	55,372,299
Total	**258,510,150**

Table 2.1: Distribution of AIS signals among dry-bulk sub-segments in the AIS dataset

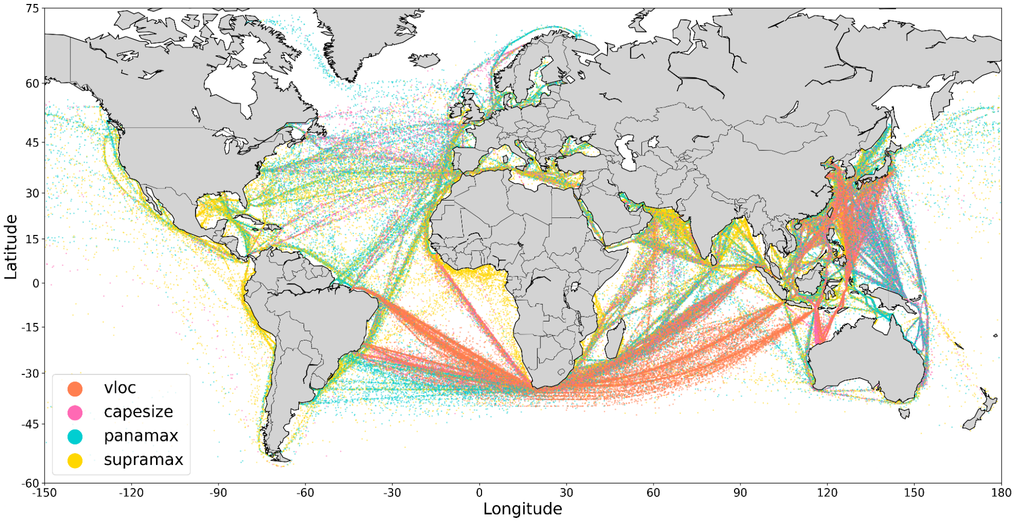


Figure 2.4: 150.000 subsampled AIS signals from the raw dataframe.

2.4.1 Trajectory definition

A trajectory is a sequence of spatial-temporal points representing the path taken by an object, such as a vessel, in a geographical space over time. In the context of maritime transportation, a trajectory between two ports can be mathematically defined as follows:

Let P_i and P_j be two distinct ports, where P_i represents the departure port and P_j represents the arrival port. The trajectory T_{ij} of a vessel traveling from port P_i to port P_j is a continuous function that maps the time interval $[t_0, t_1]$ to a sequence of spatial-temporal points:

$$T_{ij}(t) = \{(x(t), y(t), t) | t \in [t_0, t_1]\}, \tag{2.4}$$

where t_0 is the departure time from port P_i, t_1 is the arrival time at port P_j, and $(x(t), y(t))$ denotes the geographical coordinates (longitude and latitude) of the vessel at time t. The function $T_{ij}(t)$ captures the spatial-temporal evolution of the vessel's position during the voyage between the two ports.

In the supplied data, the trajectory points are sampled regularly, so a discrete representation is needed. A discrete representation of the trajectory T_{ij} can be obtained by sampling the spatio-temporal points at regular time intervals during the voyage. These sampled points can be represented as:

$$T_{ij}^D = \{(x(t_k), y(t_k), t_k) | t_k = t_0, t_0 + dt, \ldots, t_1\}, \tag{2.5}$$

where T_{ij}^D is the discrete representation of the trajectory, and the sampling interval is determined by discretization parameter dt. From here on, T_{ij}^D will be simplified as T_{ij}.

2.4.2 Trajectory re-sampling

Re-sampling on time

AIS signals are transmitted at high-frequency intervals, often resulting in a large volume of trajectory data. The provided data has signals provided at every hour. In order to effectively use this information for predicting the destination of vessels, it can be helpful to re-sample the trajectory data to reduce noise and computational complexity. Re-sampling has been done in several similar studies, such as in Omholt-Jensen (2021), where the re-sampling was done based upon six hour time intervals.

Re-sampling trajectory data has several benefits for AIS-based destination prediction:

1. **Noise reduction:** High-frequency AIS signals can contain noise due to various factors such as signal interference, transmission errors, or changes in environmental conditions (Poļevskis et al. 2012). By re-sampling the data at lower frequencies, the impact of noise on the prediction model can be reduced, leading to more accurate and reliable predictions.
2. **Computational efficiency:** The large volume of high-frequency AIS data can pose significant challenges for the storage and processing of trajectory information. Re-sampling the data allows for a more manageable dataset size, reducing the computational resources required for calculating distances between trajectories.
3. **Generalization:** Re-sampling trajectory data at lower frequencies can help in capturing the general trends and patterns in vessel movements, making the prediction model more robust to local variations and outliers in the data. This generalization can lead to improved performance in predicting destination, discharge or loading ports.

Using the above points, several re-sampling distances were investigated. An example of re-sampling the data at intervals of six hours is displayed in Figure 2.5. In the figure, one can clearly see how the shape of the trajectory remains the same, but much less cluttered. This re-sampling time was also chosen in this thesis, due to the volume of the data and the previous approaches gaining effective results with this re-sampling. This gives a discretization parameter $dt = 6$ for the trajectories, since the AIS signals are supplied hourly.

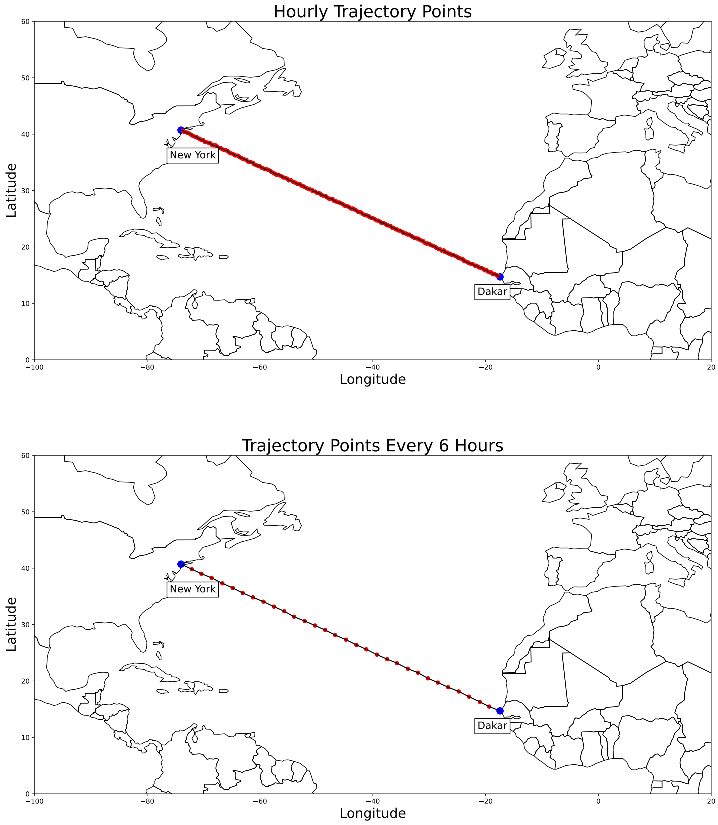
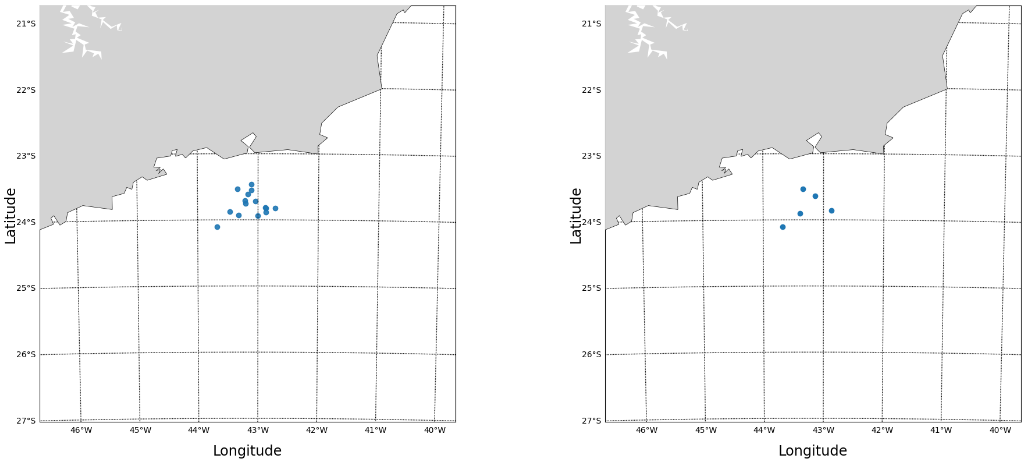


Figure 2.5: Example voyage traveling from Dakar to New York. The top figure is for AIS signals sampled every hour, and the bottom figure is the data re-sampled at intervals of six hours.

Re-sampling on distance

As vessels move through different areas, their positions can be reported at varying intervals, leading to an uneven distribution of data points. It is common for vessels to bunker outside of a port before being able to enter for loading, or discharging cargo. Re-sampling based upon time can address this somewhat, but can still lead to an uneven distribution. Another way to address this issue is by re-sampling the trajectory data based on distance.

Re-sampling based on distance involves combining data points that are closer to each other than a specified threshold. This process ensures that the trajectory data is uniformly distributed across the spatial domain, allowing for more similar trajectories, since clustered trajectory points would only make the trajectory calculations more noisy. An example is displayed in Figure 2.6. If a vessel travels less than 10 km during 6 hours, it has a speed of 0.899 knots, which essentially means that the ship is at rest, therefore 10km is chosen as the threshold for re-sampling the vessel trajectory data.



(a) Original **(b)** Re-sampled on distance

Figure 2.6: AIS signals from a ship outside Rio de Janeiro: Original and re-sampled and distance.

2.4.3 Temporal voyage segmentation

Port-to-port segmentation

When predicting destinations, a key challenge arises from the fact that for new, unseen voyages, we do not always know how far they have progressed towards their destination.To handle this uncertainty and to make the model more robust and adaptive, the trajectories for all voyages are split into four distinct temporal stages based on the time remaining to arrival at the destination port:

- **Stage 1**: Voyages with 1 day remaining to destination.
- **Stage 2**: Voyages with 2 days remaining to destination.
- **Stage 3**: Voyages with 3 days remaining to destination.
- **Stage 4**: Voyages with 1 week remaining to destination.

This is done in other similar studies, such as in Zhang et al. (2020) and Omholt-Jensen (2021). Each stage represents a different phase of the voyage. If the voyage is shorter than the temporal divide, only the ones that fit are kept. Different patterns may emerge at different stages of the voyage, which could be beneficial in predicting the destination. An example of this is displayed in Figure 2.7, a voyage between Port Walcott in Australia and Zhanjiang in China. The voyage is divided into different colors, where at the split between purple and green, the voyage has one week left, between green and blue the voyage has three days left, between blue and red the voyage has two days left, and between red and gray the voyage has one day left.

In the figure, it becomes clear that predicting the destination using AIS signals would be more difficult for the vessel at one week away from the arrival port, because of the large amount of possible destinations still ahead. When there is only one day left, one can disregard arrival ports in e.g the Philippines or Japan, since that would not make sense considering the trajectory and current position. Voyages that are closer than one day in time, so the gray area in the figure, are not included in the dataset,

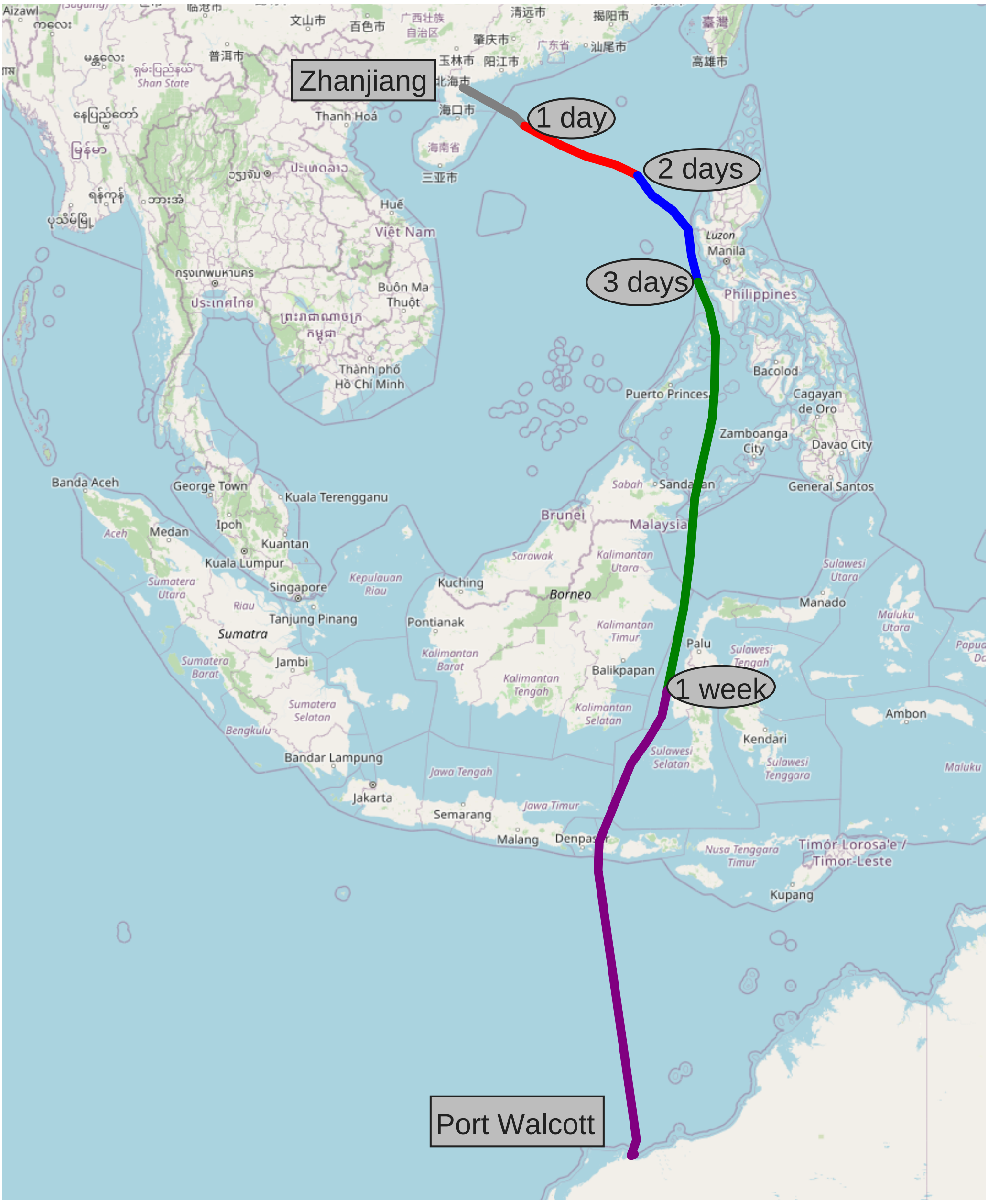


Figure 2.7: Voyage between Port Walcott and Zhanjiang, temporally divided.

since prediction of these is not interesting since they are already so close to their destination.

For each voyage, we generate a separate sample for each temporal group it belongs to. For instance, a voyage with an actual duration of 8 days will have a record in each of the four stages. Each sample contains the state of the voyage as it would be t days before arrival, where t is the number of days corresponding to the group.

By doing so, we effectively augment our dataset, which allows the machine learning model to learn the patterns corresponding to different stages of the voyage. The entire augmented dataset, combining all four groups, is then used to train the multi-class classifier. In the prediction phase, for a new, unseen voyage, we can use the current voyage data and the trained classifier to predict the destination, regardless of how far the vessel has progressed in its journey. This ensures also that there is no data leakage between stages in the prediction phase.

Laden and ballast time segmentation

The laden voyages can be significantly longer than the port-to-port voyages, since they can have several port stops during the way. Also, each laden and ballast voyage start and stop with a significant event. Therefore a different approach, which mimics a real-world scenario is introduced for laden and ballast voyages, where the temporal split is based upon days and weeks after departure of the starting port, which is the load port for laden voyagesm and the unload port for ballast voyages. This split is done up to 5 weeks:

- **Stage 1**: Voyages 2 days after leaving the load port.
- **Stage 2**: Voyages 4 days after leaving the load port.
- **Stage 3**: Voyages 1 week after leaving the load port.
- **Stage 4**: Voyages 2 weeks after leaving the load port.
- **Stage 5**: Voyages 3 weeks after leaving the load port.
- **Stage 6**: Voyages 4 weeks after leaving the load port.

- **Stage 7**: Voyages 5 weeks after leaving the load port.

This provides a more realistic method measure predictability, since this information is readily available for unseen voyages.

2.4.4 Full trajectory creation process

The entire process of creating trajectories can be described by

1. **Select relevant columns**: Filter the dataset to retain only the relevant columns for further analysis.
2. **Create trajectories**: Use departure port, destination port, and departure time to create trajectories for each voyage. These trajectories represent the chronological sequence of geographical positions of a vessel during a trip. For the load-to-unload dataset, loadport and unloadport are used.
3. **Re-sample trajectories on a 6-hour basis**: Replace data with mean values for every 6-hour interval. This step ensures uniformity in the time intervals between data points across all trajectories.
4. **Re-sample based on distance**: If two consecutive points in a trajectory are less than a certain threshold (e.g., 10 km) apart, combine these points. This step aims to remove excessive points in areas where vessels are stationary or moving within a confined region.
5. **Filter out outliers and NaNs**: Remove anomalous data points and data points with missing values (NaNs) to ensure the quality of the dataset.
6. **Temporally divide trajectories**: Split trajectories based upon time.
7. **Filter out low-frequency ports**: Exclude voyages that depart from or arrive at ports with low occurrence frequencies. This step can be achieved by filtering out all voyages from or to ports with fewer than a certain number of other departures or arrivals. By doing so, the dataset will focus on more common ports, thereby increasing the model's predictive power for more typical cases. This threshold is set to 20 after inspection of the data set.

After the full process, the signals displayed in Figure 2.4 are transformed into trajectories such as in Figure 2.8.

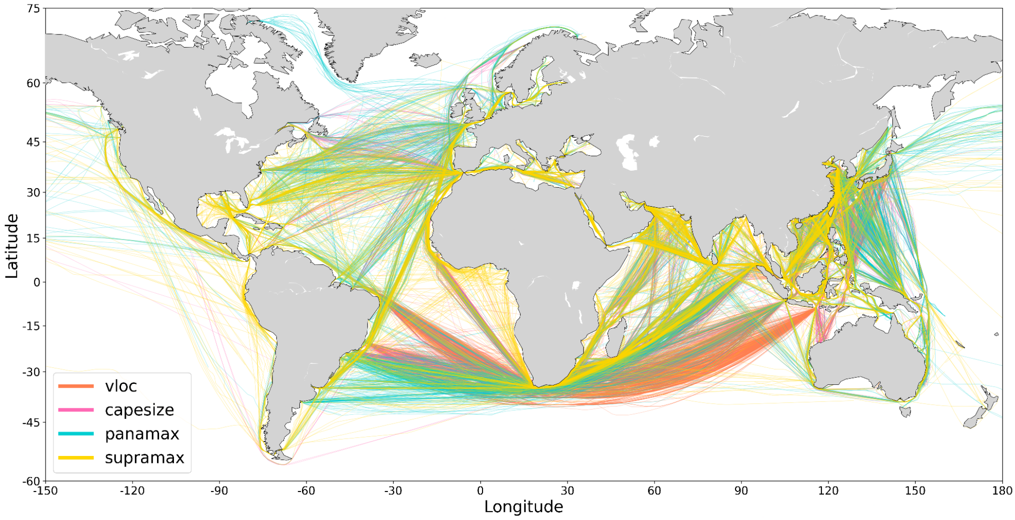


Figure 2.8: 50.000 subsampled trajectories from the dataset, equally split between sub-segments.

2.5 Port clustering

In this section, the Haversine formula is introduced, along with a method for reducing the response search space called port clustering.

2.5.1 Haversine formula

The Haversine formula is used to calculate the great-circle distance between two points on the surface of a sphere, given their longitudes and latitudes. It is particularly useful for calculating distances between geographical coordinates on Earth. The Haversine distance between two points $p_1 = (lat_1, lon_1)$ and $p_2 = (lat_2, lon_2)$ can be computed as follows:

$$d_{\text{H-sine}}(p_1, p_2) = 2R \arcsin \sqrt{\phi(\text{lat}_2, \text{lat}_1) + \cos(\text{lat}_1)\cos(\text{lat}_2)\phi(\text{lon}_2, \text{lon}_1)}, \qquad (2.6)$$

where $\phi(a, b) = \sin^2(\frac{a-b}{2})$, R is the Earth's radius (approximately 6371 km), and lat_1, lon_1, lat_2, and lon_2 are the latitudes and longitudes of the two points in radians. An illustration between two points on Earth, P and Q, is provided in Figure 2.9.

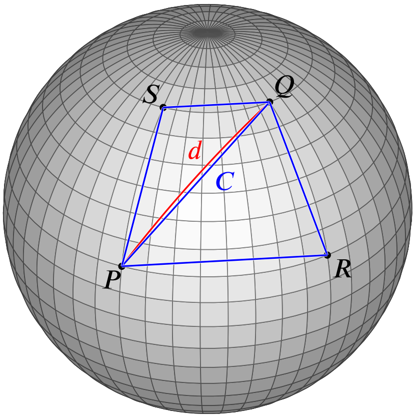


Figure 2.9: The Haversine distance between points P and Q, where the haversine distance is the red line, and the blue lines are straight chords used in the calculation, with swapped longitude coordinates.

2.5.2 Clustering

The world's shipping network is composed of thousands of ports, making it a large and complex response space. Even though larger dry-bulk vessels can not travel to all ports, the response space is still large. Given the vast number of potential destinations a vessel can have, it significantly complicates the task of predicting the destination port accurately. Many of these ports can be geographically very close to each other, leading to a high degree of overlap in their feature spaces. For instance, two ports in close proximity may have similar types of cargo, similar vessel traffic, and may serve similar types of dry bulk industries. This overlap makes it difficult to distinguish between such ports based solely on the available features.

Therefore, ports are clustered based upon their geographical proximity, and the classification problem will also be tested on this response space. The exact port within the cluster can then be predicted in a subsequent step, either using additional features or models that focus specifically on differentiating between ports within the same cluster, or by expert knowledge. This step is not performed in this thesis.

The Density-Based Spatial Clustering of Applications with Noise (DBSCAN) is used to perform the port clustering. DBSCAN (Schubert et al. 2017) is a density-based clustering algorithm which is particularly suitable for tasks where the number of clusters is not previously known, such as this one, and where the clusters may not be of the same density.

This algorithm considers clusters as regions in the dataset where the density of ports is significantly high. The necessary notation for a better understanding of the DBSCAN methodology is introduced

- $\mathcal{P} = P_1, P_2, ..., P_n$ represents the set of all ports where P_i is the location of the i-th port given in latitude and longitude coordinates.
- $\varepsilon > 0$ denotes the radius of the neighborhoods around a port.
- MinPts $\in \mathbb{N}$ is the minimum number of ports needed to form a dense region,

i.e., a cluster.

Given this, we can define the notion of ε-neighborhood of a port P_i as:

$$N_\varepsilon(P_i) = \{P_j \in \mathcal{P} : d_{\text{H-sine}}(P_i, P_j) \leq \varepsilon\}, \tag{2.7}$$

where $d_{\text{H-sine}}(P_i, P_j)$ is the haversine distance between port P_i and P_j, as defined in Equation 2.6.

A port P_i is defined as a:

- *Core port* if $|N_\varepsilon(P_i)| \geq$ MinPts,
- *Border port* if $|N_\varepsilon(P_i)| <$ MinPts but $\exists P_j : P_i \in N_\varepsilon(P_j)$ and $|N_\varepsilon(P_j)| \geq$ MinPts,
- *Noise port* otherwise.

Using these definitions, the DBSCAN algorithm can be summarized as follows:

1. For each unvisited port $P_i \in \mathcal{P}$, a neighborhood $N_\varepsilon(P_i)$ is retrieved. If it contains at least MinPts, a new cluster is created, and P_i is marked as a *core port*.
2. All ports in $N_\varepsilon(P_i)$ are added to the same cluster. If they are *core ports* as well, their neighborhoods are also added to the cluster.
3. Repeat the procedure until all ports in the dataset have been visited. The final output is a set of clusters where each cluster consists of at least MinPts within the neighborhood of ε.

The output of the DBSCAN algorithm is a set of clusters from the ports based on their geographical proximity, which should be informative in improving the accuracy of predicting the vessel's destination. The parameter MinPts is set to 1, to ensure all ports are considered. The parameter ε is more difficult to choose, since it becomes a question dependent on the user of the model. In Figure 2.10, the amount of clusters for the Capesize port destinations are displayed as a function of the intra-distance, measured with the haversine distance, within each cluster, which is the ε normalized to be kilometers.

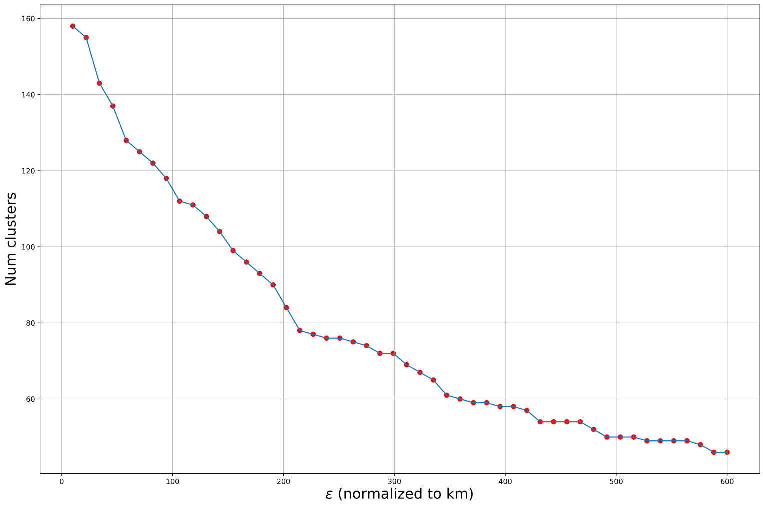
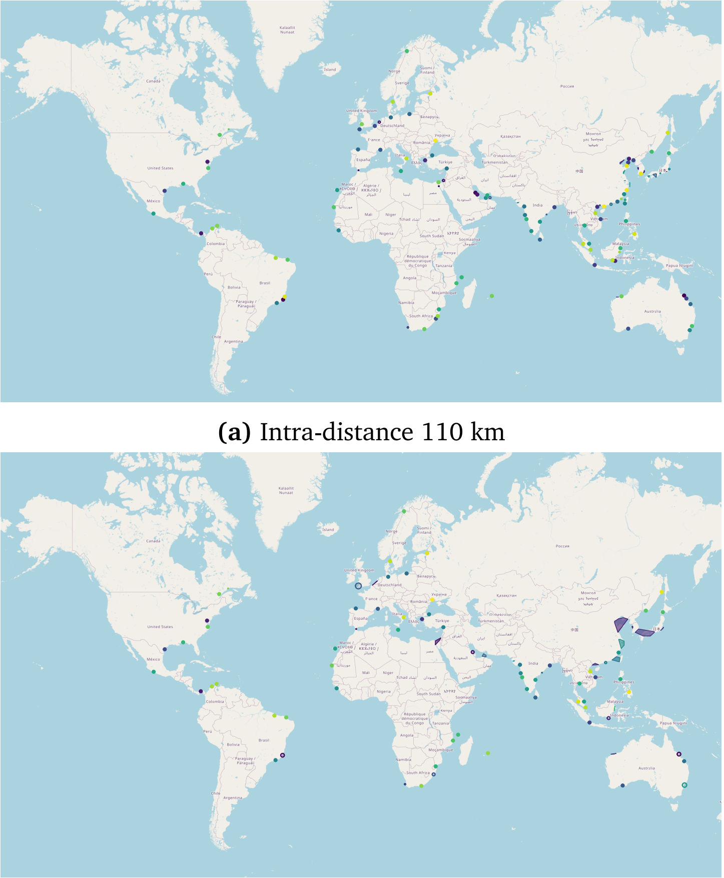


Figure 2.10: Number of port clusters for Capesize vessels for each intra-distance value used in the DBSCAN algorithm.

Based upon this graph, there seemed to be slighly larger drop off at around 110km and 220km, so the clusters for these distances were plotted on a world map with these distances. For clusters with only one port, a circle was plotted with full opacity. For clusters with two ports, a circle with half opacity was plotted, with the periphery of the circle on each port. For clusters with three or more ports, a polygon was plotted. These two different distances, for Capesize port-to-port voyages, are displayed in Figure 2.11a and Figure 2.11b, respectively.



(a) Intra-distance 110 km

(b) Intra-distance 220 km

Figure 2.11: Clustering of Capesize ports with different intra-distances.

At the first inspection the 220 km clusters seemed to cluster the ports in a well-suited manner, but after further inspection in some areas, one could see that clusters were divided across countries, which is not something AF Code and other companies wants, since calculating the imported goods into a country is of interest. An example of this is shown in Figure 2.12, east of China, where a cluster collects ports from both Japan

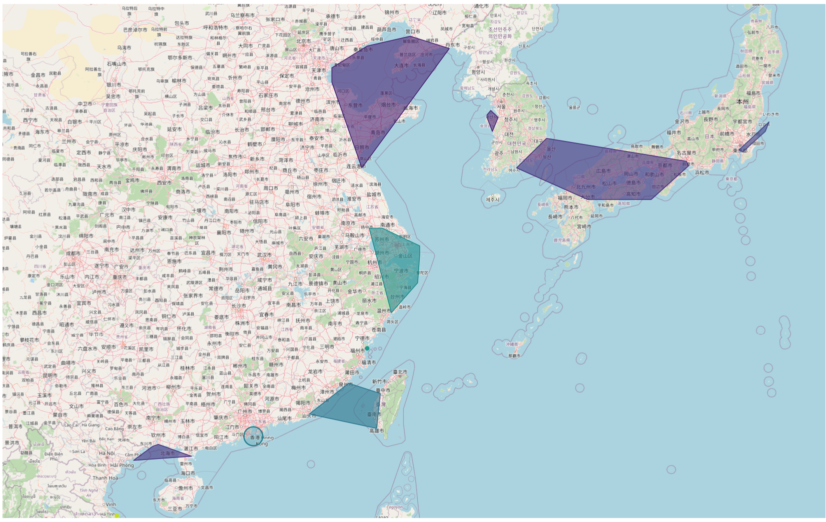
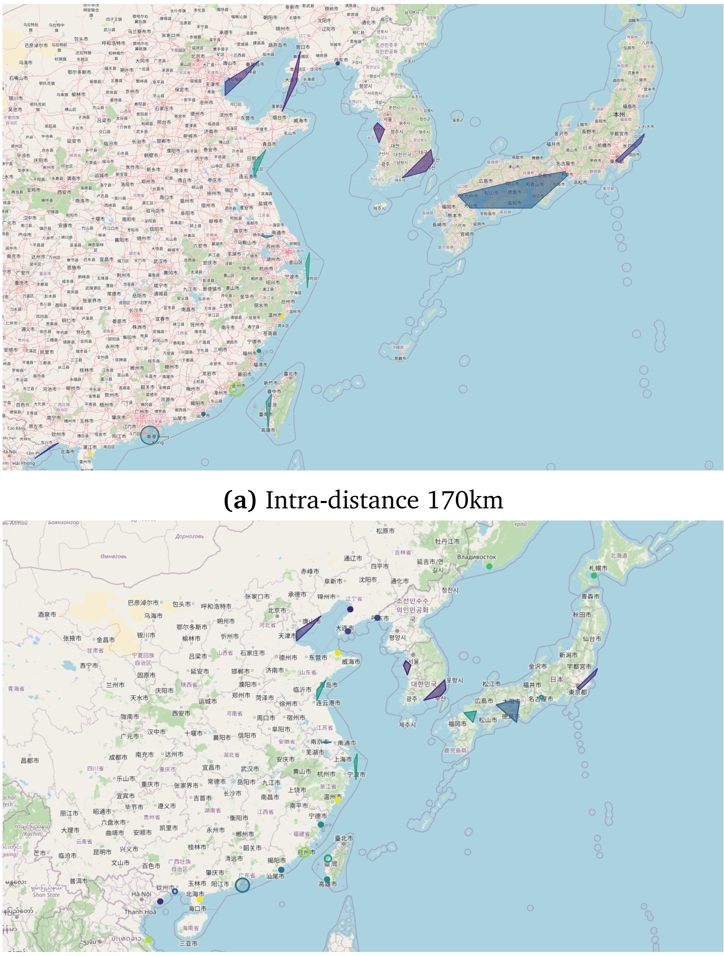


Figure 2.12: Clustering of Capesize ports with intra-distance 220km, zoomed in on eastern Asia

and South Korea, and another one between Taiwan and China.

To address these issues, alternative distances were investigated. The analysis showed that a distance of 170 km effectively fixed the first problem areas, providing a satisfactory balance between interpretability and reduction in the size of the response space. In Figure 2.13a, we illustrate the same areas at a 170 km distance, showing the clear differences from the previous figure.

However in the bottom left corner of Figure 2.12 one can see that there is a cluster that spans across the border of China and Vietnam. The distance between the responsible ports was calculated to be 129 km, so this was set as the threshold, resulting in the clusters displayed in Figure 2.13b.



(a) Intra-distance 170km

(b) Intra-distance 129km

Figure 2.13: Clustering of Capesize ports with different intra-distances, zoomed in on eastern Asia

For the Capesize ports, the number of clusters dropped to 109 when using a 129 km intra-distance, a substantial decrease from the original response space size of 158.

This provides an interesting approach by building separate models for prediction of ports and clusters for the voyages.

The process was repeated, ensuring no ports span across countries, for Panamax and Supramax ports, and then again for the three laden and ballast datasets. The response space reductions for all datasets are displayed in Table 2.2, along with the chosen intra-distance.

Table 2.2: Clustering of ports for port-to-port, laden and ballast voyages

Port-to-Port			
Sub-segment	**Unique Ports**	**Unique Clusters**	**Intra-distance(km)**
VLOC	55	26	215
Capesize	158	109	129
Panamax	230	146	120
Supramax	260	185	120
Laden journeys			
Sub-segment	**Unique Ports**	**Unique Clusters**	**Intra-distance**
VLOC	49	25	220
Capesize	82	57	120
Panamax	169	97	110
Supramax	189	120	109
Ballast journeys			
Sub-segment	**Unique Ports**	**Unique Clusters**	**Intra-distance**
VLOC	52	24	400
Capesize	88	52	350
Panamax	101	72	256
Supramax	132	76	230

2.6 Exploratory data analysis

To ensure efficient computational run time in the prediction phase, we limit the number of trajectories used to 50,000 for each sub-segment, as displayed in Figure 2.8. The laden and ballast datasets are constructed from these, resulting in smaller subdatasets. However, the exploratory data analysis is done on the full dataset.

The dataset contains voyages from February 2018 to May 2023. Significant global events could cause substantial disruption in maritime transportation patterns, and therefore conceivably impact the accuracy of our predictive models. Such events include

- The Suez Canal blockage (J. M.-y. Lee and Wong 2021), which led to that some Panamax and Supramax vessels had to re-route around Africa
- The Russia-Ukraine war, which significantly impacted the one of the largest grain exporters in Ukraine (Ngoc et al. 2022)
- The COVID-19 pandemic, which in general caused instability in the global market by extending port waiting times, but did not in general disrupt the trading routes (Michail and Melas 2020)
- The Brumadinho Dam Disaster, close to the Córrego do Feijão iron ore mine, which disrupted the Iron ore trade out of Brazil, which led to an increase in exports from smaller export countries such as India, Canada and Peru. The stock price of Vale SA, the shipping company previously mentioned that holds the Valemax fleet, plummeted and 19 billion dollars in market value were lost (Laier 2019).

However, considering the broad timeframe of the dataset (from February 2018 to May 2023), the overall impact of these specific incidents may be less important than first thought. This is because the models takes into account a diverse range of voyages over an extended period, thus smoothing out short-term anomalies and focusing more on enduring patterns. It is also worth noting that the global shipping industry is resilient and tends to adapt to such disruptions over time (Chua et al. 2022). However, these events underscore the need for models to be adaptable and responsive to the ever-changing dynamics of global shipping.

In the dataset, the average travel distance of a port-to-port voyage is 1878 km, and the average travel duration is 145 hours. The dataset provides several expected patterns about the port visitation in the global maritime industry, as illustrated in Figure 2.14. Here one can see a clear imbalance in visitation and also the sheer scale of the amounts of ports. Note that the maximum count of the largest loadport is larger than the largest departure port count, since the first leg of some laden voyages may not be present in the dataset due to data cleaning.

According to the distributions of voyage origins and destinations in Figure 2.15, Singapore (SGP), China (CHN), and Australia (AUS) dominate as the main nodes in the shipping network across different vessel sub-segments. We note a significant concentration of traffic in Eastern Asia, which is consistent with the region's dominant role in the dry bulk industry. Countries in this region, such as China, Japan, and South Korea, are known for their substantial involvement in global trade, whether through exporting manufactured goods or importing raw materials (Jin et al. 2006). Singapore is particularly significant as a voyage starting point, especially for the VLOC sub-segment, where more than 11,900 voyages start from Singapore. Australia, primarily via Port Hedland, is also a prevalent departure point, again especially for VLOC sub-segment. China, on the other hand, tends to be a popular destination. For the VLOC sub-segment, over 20,276 voyages ended in China. For the other sub-segments, the spread of departure ports is more even, which aligns with the belief that VLOC vessels tend to stick to larger ports. This is seen in Table 2.2, where VLOC has a maximum of 55 unique ports, while the other sub-segments have many more.

The data also reveals that the busiest ports generally align with the most active countries. However, there are exceptions. For instance, while Brazil (BRA) ranks among the top three countries for VLOC voyage origins, its ports do not appear in the top 15 departure ports for this sub-segments. This discrepancy might be due to the fact that voyages from Brazil might be distributed among several ports, with none individually breaking into the top 15. Furthermore, the distribution of voyages between vessel types provides valuable insights. Singapore, for example, appears as a major

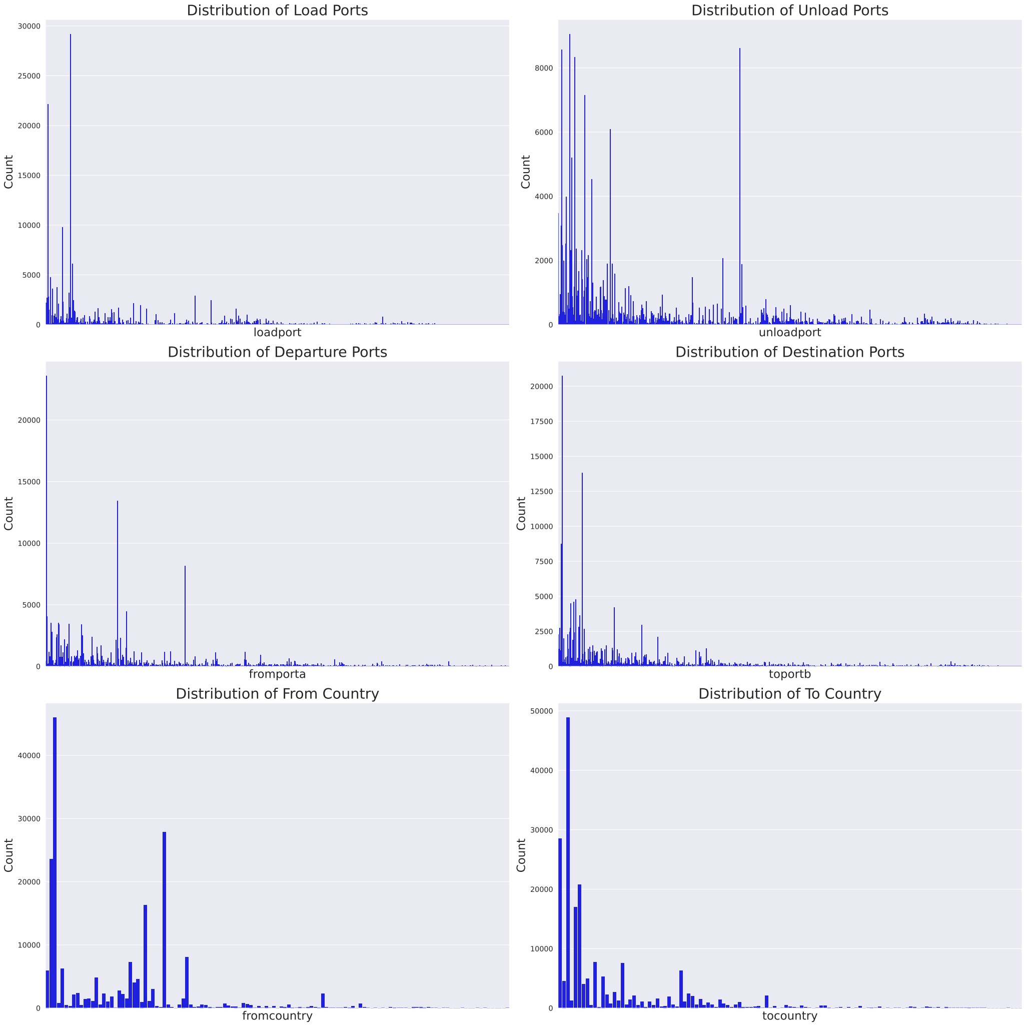


Figure 2.14: Distribution of destination and departure ports and countries, and load and unloading ports, in the dataset.

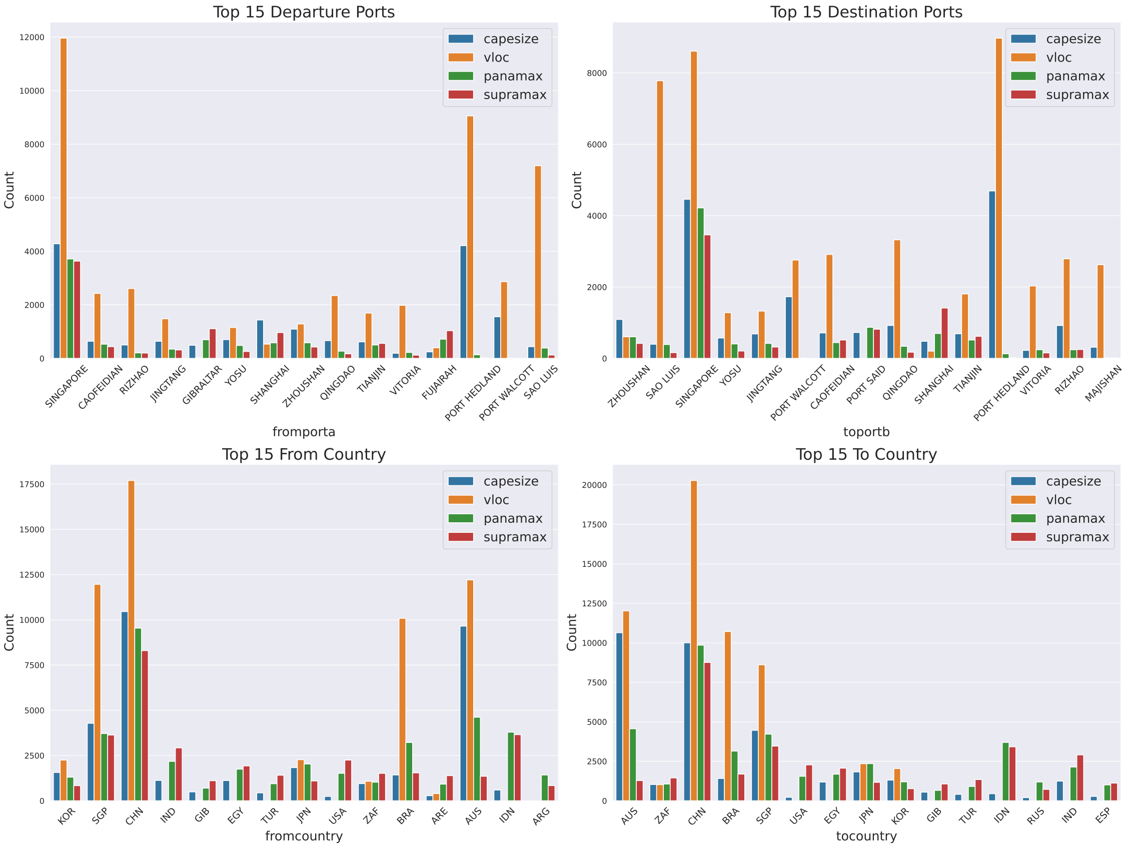


Figure 2.15: Top 15 distribution of arrival and departure ports and countries in the dataset.

hub across all vessel types, underlining its status as a global maritime center. These observations provide a snapshot of global shipping patterns and highlight the critical role of certain countries and ports.

Figure 2.16 displays the distribution of arrival and departure ports plotted on a word map, where one can see the clear clustering of ports up in Eastern Asia. Figure 2.17 shows the top 100 for the same distributions, showing more clearly where most of the traffic is. Notably, the density of traffic in Europe and on the east coast of the United States is significantly lower compared to Asia. This disparity underscores the dominant role of Asian ports in global maritime trade and transportation.

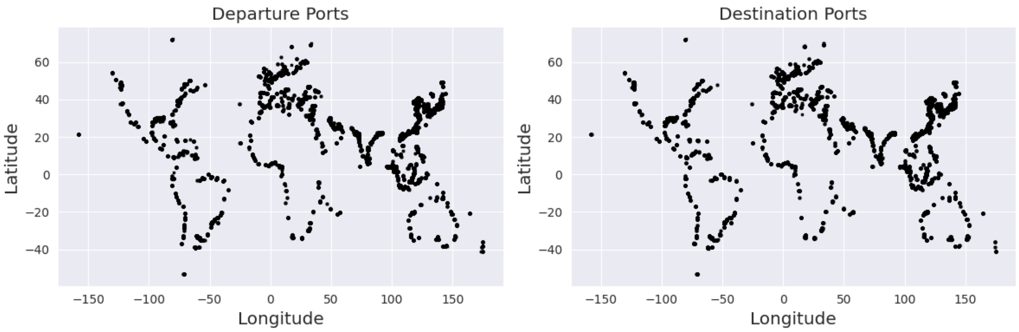


Figure 2.16: Latitude and longitude distribution of ports in the dataset.

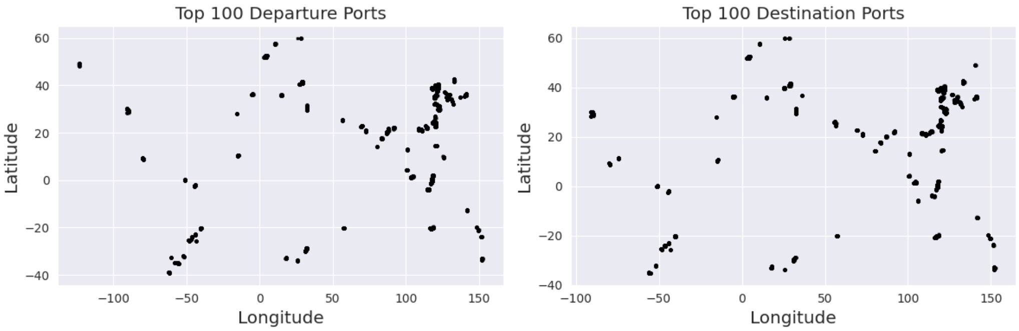


Figure 2.17: Latitude and longitude distribution of top 100 departure and destination ports in the dataset.

This is imbalance can also clearly be seen if one displays the count of destination ports within each country on a heatmap on the world. The scale is logarithmic, showing clearly that China and Singapore(combined with Malaysia to easier display it) are the two main hubs, and that Australia and Brazil are also quite frequent. The map also shows the variation in Europe, where for instance Sweden has very few vessel arrivals compared to Norway, which makes sense considering that even though Sweden has significant iron ore mining in Kiruna, most of the iron ore is shipped out of Narvik, Norway (Carlson 1953).

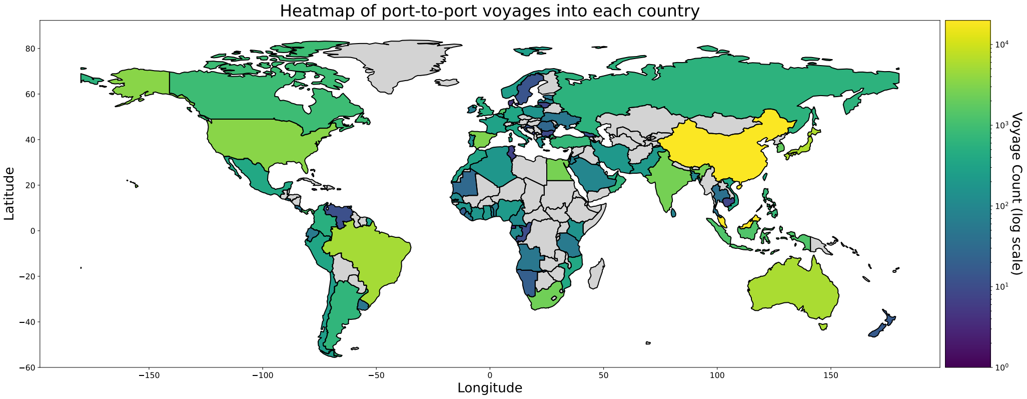


Figure 2.18: Heatmap of the distribution of destination ports, grouped by country. The scale is logarithmic to capture the nuances more easily. Note that Singapore and Malaysia are combined into one.

Figure 2.19 displays the distribution of the temporal segmentation of the voyages, along with the distribution of the sub-segments. There are fewer and fewer voyages for each day before arrival, indicating that there are more short voyages than long.

To illustrate how the laden voyages operate between countries, a sankey diagram is displayed in Figure 2.20. It shows the flow of trade between countries, with a threshold of at least 100 voyages between two countries. Here one can see that the majority of laden voyages go from Australia and China, and also that Brazil and South Africa(ZAF) are important factors.

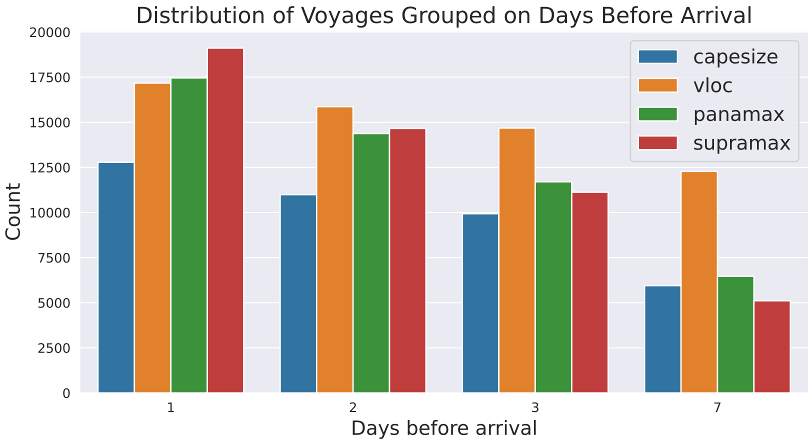


Figure 2.19: Distribution of voyages for days before arrival.

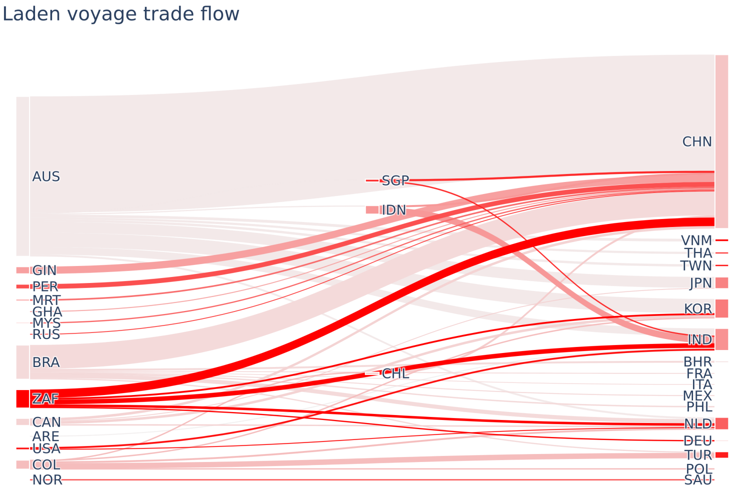


Figure 2.20: Flow of the laden voyages, on a threshold of at least 100 voyages between two countries.

Chapter 3

Methodology

3.1 Multiclass classification problem notation

In this section, we introduce the notation for the multiclass classification problem, where the goal is to predict vessel destinations and discharge destinations using a set of features. The problem can be described using the following notation

- Let $\mathbf{X} \in \mathbb{R}^{n \times p}$ be the feature matrix, where n represents the number of observations (vessel voyages) and p denotes the number of features. Each row $\mathbf{x}_i$ of the matrix corresponds to a feature vector for the i-th vessel voyage, with $i = 1, 2, \ldots, n$. For each port-to-port dataset,$n = 50000$, and for laden and ballast, $n < 50000$.
- Let $\boldsymbol{y} \in \mathcal{Y}^n$ be the response vector, where $\mathcal{Y} = \{1, 2, \ldots, K\}$ denotes the set of possible destination classes, and K represents the total number of classes for the particular sub-segment. The element y_i of the vector corresponds to the true destination class for the i-th vessel voyage.
- Let $\hat{\boldsymbol{y}} \in \mathcal{Y}^n$ be the predicted response vector, where $\hat{y}_i$ corresponds to the pre-

dicted destination class for the i-th vessel voyage.

The objective of the multiclass classification problem is to learn a function $f : \mathbb{R}^p \to \mathcal{Y}$ that maps the feature vector $\mathbf{x}_i$ to the corresponding destination class y_i for each vessel, minimizing the classification error, a suitable loss function or other relevant metrics.

In the simplest case, let $\mathbf{x}_i$ denote the departure port for the i-th vessel voyage. For each departure port, we could use the count from that port to each possible destination $y \in \mathcal{Y}$ as a predictor, where one selects the prediction as the destination with the highest count.

Another possible approach to the multiclass classification problem would be to use logistic regression. In this case, we learn a function $f : \mathbb{R}^p \to \mathcal{Y}$ that maps a feature vector $\mathbf{x}_i$ to the corresponding destination class y_i. This feature vector could include both quantitative features (like the counts described above) and qualitative features (like the vessel type or the departure port).

The multiclass logistic model can be represented as:

$$\log\left(\frac{P(Y=k|\mathbf{x})}{P(Y=K|\mathbf{x})}\right) = \boldsymbol{\beta}_{0k} + \boldsymbol{\beta}_k^T\mathbf{x}, \tag{3.1}$$

for $k = 1, 2, \ldots, K-1$ and $\mathbf{x} \in \mathbb{R}^p$, where $P(Y = k|\mathbf{x})$ is the probability of the k-th destination given the feature vector $\mathbf{x}$, $\boldsymbol{\beta}_{0k}$ is the intercept for the k-th class, and $\boldsymbol{\beta}_k$ is the vector of coefficients for the k-th class.

3.2 Trajectory-based features

When predicting destinations, the trajectory of a voyage represents information that can be very important for increasing the accuracy of a prediction. The trajectory of a vessel, which is essentially the path it follows over time, is determined by a combination of factors including the vessel's starting point, its destination, and the conditions it encounters during the voyage. As such, extracting features from these trajectories can significantly enhance the predictive power of our model.

3.2.1 Hausdorff distance with Haversine formula for trajectory comparison

The Hausdorff distance is a measure used to quantify the similarity between two sets of points. In the context of this thesis and maritime transportation, the Hausdorff distance can be used to compare two trajectories originating from the same departure port but terminating at different arrival ports. To account for the Earth's spherical geometry, we will use the Haversine distance as the underlying metric.

Given two trajectories, T_{i1} and T_{i2}, originating from the same departure port P_i, but terminating at different arrival ports P_1 and P_2, the directed Hausdorff distance from trajectory T_{i1} to trajectory T_{i2} is defined as:

$$d_{Hdir}(T_{i1}, T_{i2}) = \max_{p \in T_{i1}} \min_{q \in T_{i2}} d_{\text{H-sine}}(p, q). \tag{3.2}$$

Here, T_{i1} and T_{i2} represent the trajectories originating from port P_i and terminating at ports P_1 and P_2, respectively.

The directed Hausdorff distance measures the maximum distance from a point p on T_{i1} to its closest point q on T_{i2}, and it is calculated using the directed Hausdorff sine distance, $d_{\text{H-sine}}(p,q)$. An illustration is displayed in Figure 3.1.

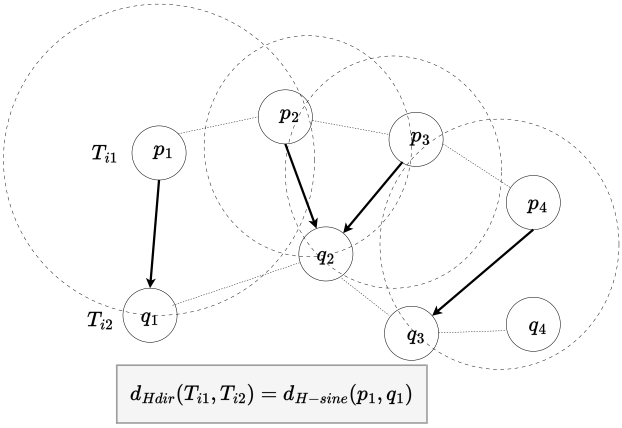


Figure 3.1: The directed Hausdorff Distance between trajectories T_{i1} and T_{i2}. The distance function can be manually selected, but due to the nature of this thesis, the Haversine distance is chosen.

Similarly, the directed Hausdorff distance from trajectory T_{i2} to trajectory T_{i1} is:

$$d_{Hdir}(T_{i2}, T_{i1}) = \max_{p \in T_{i2}} \min_{q \in T_{i1}} d_{\text{H-sine}}(p,q). \tag{3.3}$$

Finally, the Hausdorff distance between the two trajectories, T_{i1} and T_{i2}, is the maximum of the directed Hausdorff distances:

$$d_H(T_{i1}, T_{i2}) = \max d_{Hdir}(T_{i1}, T_{i2}), d_{Hdir}(T_{i2}, T_{i1}). \tag{3.4}$$

By calculating the Hausdorff distance with the Haversine metric, we can compare two trajectories with the same departure port but different arrival ports, taking into account the curvature of the Earth. An illustration is displayed in Figure 3.2

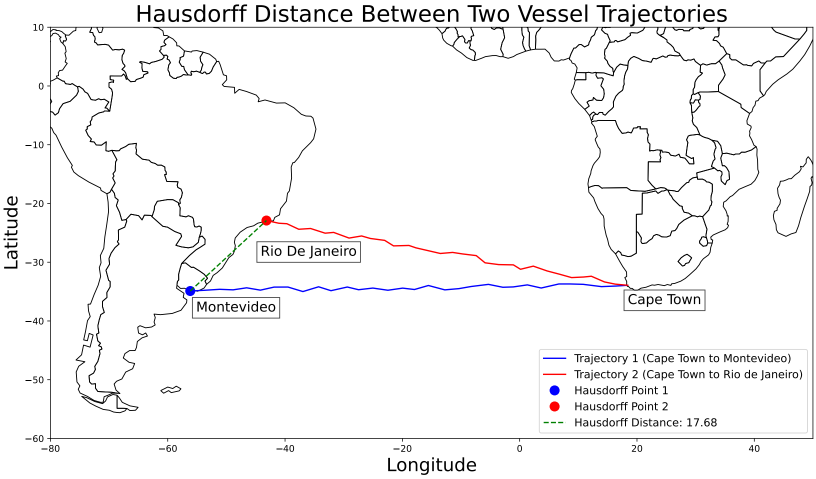


Figure 3.2: The directed Hausdorff Distance between two trajectories.

There exists a python library (Taha and Hanbury 2015), which implements the Hausdorff distance between coordinates in a fast way.

3.2.2 Symmetrized Segment-Path Distance with Haversine formula for trajectory comparison

In this section, another distance metric called Symmetrized Segment-Path Distance (SSPD) is introduced (Besse et al. 2016). This distance has been used in previous similar works (Omholt-Jensen 2021). SSPD is proposed as a shape-based distance that takes into account the whole trajectories and is less affected by noise than other distance measures.

The distance D_{pT} from a point p to a trajectory T is the minimum of distances between this point and all points q that compose T, similar to the above

$$D_{pT}(p,T) = \min_{q \in T} d_{\text{H-sine}}(p,q). \tag{3.5}$$

The Segment-Path distance(SPD) from trajectory T_{i1} to trajectory T_{i2} is the mean of all distances from points composing T_{i1} to the trajectory T_{i2}. Thus, the SPD distance between two trajectories T_{i1} and T_{j2}, is defined as

$$D_{SPD}(T_{i1}, T_{i2}) = \frac{1}{n_1} \sum_{k_1=1}^{n_1} D_{pT}(p^1_{k_1}, T_{i2}), \tag{3.6}$$

where n_1 is the amount of points in T_{i1}.

The distance in Equation 3.6 is not symmetric. If T_{i1} is a very small sub-trajectory of T_{i2}, then $D_{SPD}(T_{i1}, T_{i2}) = 0$, but then $D_{SPD}(T_{i2}, T_{i1})$ can be very large. By taking the mean of these distances, the "Symmetrized Segment-Path Distance", SSPD, is defined and is symmetric.

$$D_{SSPD}(T_{i1}, T_{i2}) = \frac{D_{SPD}(T_{i1}, T_{i2}) + D_{SPD}(T_{i2}, T_{i1})}{2}. \tag{3.7}$$

An illustration of the SSPD distance between two trajectories between respectively Cape Town and Montevideo and Cape Town and Rio De Janeiro, can be seen in Figure 3.3.

Note that if one instead uses the maximum instead of the mean, this becomes the Hausdorff function described above. Since SSPD considers the mean, the distance becomes less sensitive to noise in the trajectory data. For both distance calculations, is is assumed that the departure port i is the same for both trajectories, however this is not necessary for the calculations, but is used since this thesis will only compare trajectories departing from the same port.

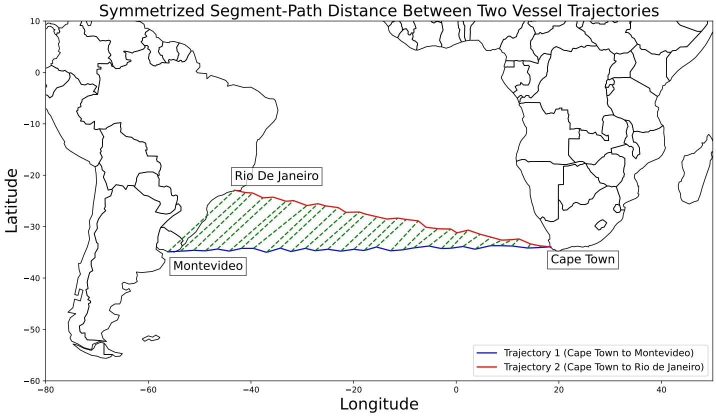

Symmetrized Segment-Path Distance Between Two Vessel Trajectories
Rio De Janeiro
Montevideo
Cape Town
Trajectory 1 (Cape Town to Montevideo)
Trajectory 2 (Cape Town to Rio de Janeiro)
Latitude
Longitude
10
0
−10
−20
−30
−40
−50
−60
−80
−60
−40
−20
0
20
40

Figure 3.3: SSPD distance between two realistic trajectories. The calculated distance is the mean.

3.2.3 Most likely trajectory destination calculation

The Hausdorff distance and SSPD distance with the Haversine formula can be used as a feature in machine learning models for destination prediction, as well as serve as an initial spatial prediction. By comparing the current trajectory of a vessel with historical trajectories, we can extract valuable information on the vessel's most likely destination. This method is inspired by Omholt-Jensen (2021). The process can be described as follows:

1. Choose a randomly sampled subset of voyages departing from the same port as historical trajectories.
2. Compute the chosen similarity measure with the Haversine formula between the current trajectory and each selected historical trajectory.
3. Select the most similar historical trajectory:
   a. Look at the ϕ most similar trajectories using SSPD or Hausdorff. The most similar is the trajectory with the lowest similarity measure value.
   b. Select the majority vote as the most likely trajectory destination
   c. Calculate the mean distance, from the similarity measure distances, to

each likely trajectory, a maximum of ϕ distances.

4. Use the destination port of the selected trajectory as the candidate destination, the Most Likely Trajectory Destination(MLTD).

An example of this is displayed in Figure 3.4, where a vessel, in orange, travels from Gladstone to an unknown location. The trajectory is compared to the other trajectories, and the three most similar are displayed in blue. The MLTD of this voyage would then be the mode of the destinations of these trajectories.

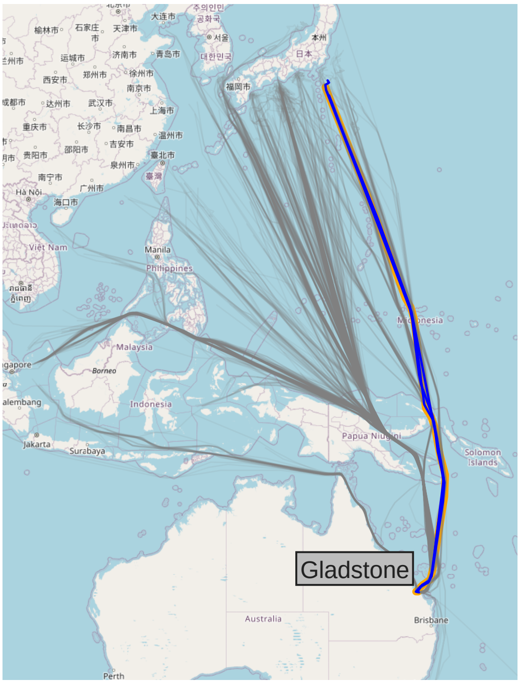


Figure 3.4: A selected voyage in orange, traveling from Gladstone. The three(thesis uses 10 as value for robustness) most similar trajectories in blue, and the MLTD is selected as the most likely destination of the destinations of the most similar trajectories.

3.3 Machine learning datasets

The full pre-processing of the trajectories for port-to-port prediction is displayed in Figure 3.5. The arrows indicate the flow of the process, where each box corresponds to a different step in the process, numbered for clarity.

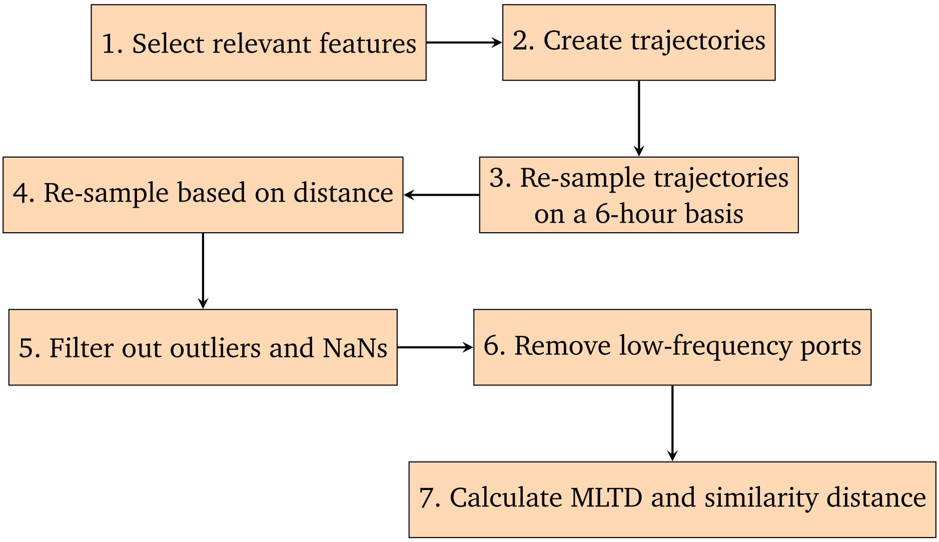

1. Select relevant features
2. Create trajectories
3. Re-sample trajectories on a 6-hour basis
4. Re-sample based on distance
5. Filter out outliers and NaNs
6. Remove low-frequency ports
7. Calculate MLTD and similarity distance

Figure 3.5: Flowchart of the process for creating the machine learning dataset

While AIS provides key trajectory-based features, such as the MLTD and coordinate features, its exclusive use for predicting destination ports is limiting. The trajectory-based features do not account for the frequency of travels between ports, information regarding the vessel such as owner, flagcode, name of departure port and other similar attributes. Certain destinations may be more common for vessels departing from Port Hedland than from Port Walcott, so in order for a statistical model to take account of these factors, they need to be included as features. AF Code has data for each unique IMO, summarized in the static list below. The importance of these features for the model is hard to quantify before training, but can be assessed through feature importance methods. All available features are detailed below, categorized into static and trajectory-based features:

3.3.1 Static features

Static features are properties of a vessel that remain constant during its journey. These attributes provide insights into the physical characteristics of the vessel and administrative information, which could influence potential destinations. For instance, larger vessels may be limited to ports that can accommodate their size, and vessels built in certain countries may have increased likelihood of traveling to specific countries due to trade agreements or economic relationships. According to AF Code, charterer could also have been an important feature, but this information is kept secret. The static features include:

- **fromporta**: The origin port of the vessel, with country available as well
- **loadport**: The loading port of the vessel, with country available as well
- **alat, alon**: Latitude and longitude of the vessel's origin port
- **breadth, length**: The dimensions of the vessel
- **deadweight**: The maximum weight that the vessel can safely carry
- **depth**: The vessel's depth, which is the vertical distance between the waterline and the bottom of the hull (keel)
- **flagcode**: The flag code the vessel is flying under
- **shipmanager, registeredowner, groupbeneficialowner**: These represent the ship manager, the registered owner of the vessel, and the group that benefits from the ownership of the vessel, respectively
- **shipbuilder**: The company that built the vessel
- **countryofbuild**: The country where the vessel was built

3.3.2 Trajectory-based features

Trajectory-based features vary during a vessel's journey. These include the vessel's current location, total distance traveled, compass direction, and other attributes related to the Most Likely Trajectory Distance (MLTD). These attributes can provide information about the vessel's current trip, location, and direction, which can be highly predictive of its destination. The trajectory-based features include:

- **MLTD**: These represent ports based on Most Likely Trajectory Distance (MLTD). The exact categories these represent will depend on how the MLTD was split into groups
- **end lat, end lon**: The final latitude and longitude of the vessel's given trajectory
- **travel distance**: The total distance travelled by the vessel so far
- **direction**: The vessel's compass direction
- **similarity distance**: The distance calculated between the trajectory of the vessel and the mean of the MLTD trajectories
- **MLTD lat, MLTD lon**: The latitude and longitude corresponding to the vessel's MLTD

The available target variables for port-to-port voyages are:

- **toportb**: The destination port of the vessel
- **cluster**: The cluster belonging to the destination port of the vessel
- **tocountry**: The country of the destination port of the vessel

For laden and ballast, the target becomes 'next_unloadport' and 'next_loadport', respectively. All categorical features are encoded using label encoding, explained in the next section.

3.4 Encoding of categorical features

Since the target variable in the prediction will be categorical, along with several other categorical features, these will need to be handled, with the two main possibilities being one-hot encoding or label encoding. Other encoding methods also exist, however exploration of these did not increase the results of this work in a significant manner.

3.4.1 One-Hot encoding

One-hot encoding(OHE) is a technique that transforms a categorical feature with K distinct classes into K binary features, each representing one class. Each new binary feature takes the value 1 if the original feature's value corresponds to the respective class, and 0 otherwise. This method results in a sparse representation, where each instance has a single 1 and the remaining values are 0s. There are however a few potential issues with OHE:

- High-dimensionality: When dealing with categorical features with a large number of classes, one-hot encoding can significantly increase the dimensionality of the dataset, leading to increased memory requirements and longer training times.
- Inefficiency: The sparse nature of one-hot encoding can result in inefficient use of memory and computational resources during model training, as most of the values are 0.

In summary, when we apply one-hot encoding to a categorical variable, it can introduce sparsity into the dataset, which is generally considered undesirable. From the perspective of the splitting algorithm in a tree ensemble method, which will be used later, each dummy variable created from the one-hot encoding is treated as an independent feature. If the decision tree algorithm decides to split on one of these dummy variables, the resulting gain in purity or information gain is typically minimal. Consequently, the decision tree is unlikely to select any of the dummy variables as important features near the root of the tree. This behavior stems from the fact that the dummy variables have a sparse representation and do not contribute significantly to the overall decision-making process.

3.4.2 Label encoding

Label encoding is a technique that assigns an integer value to each distinct class in a categorical feature. The values typically range from 0 to $K-1$, where K is the number of classes. Unlike one-hot encoding, label encoding results in a dense representation, with a single numerical value representing each instance of the original feature.

Label encoding introduces an arbitrary ordinal relationship between the classes, which may not reflect the true relationship between them. This can cause issues with some models that assume a meaningful ordering of the input features. However, due to the issues with one-hot encoding, label encoding is used for the categorical features in this thesis.

3.5 Machine learning models

3.5.1 Tree classification

Tree-based classifiers are a type of machine learning model that use decision trees to make predictions about the target variable (Quinlan 1996). Decision trees are a series of nodes that represent a decision based on a feature of the data, with each node having two or more branches that lead to other nodes or leaf nodes. Leaf nodes represent a prediction of the target variable. An example of a tree for a binary classification problem of predicting between Shanghai and Yosu as destination ports is illustrated in Figure 3.6, together with a mock feature vector $\boldsymbol{X}$, e.g being SSPD distance and deadweight of a ship.

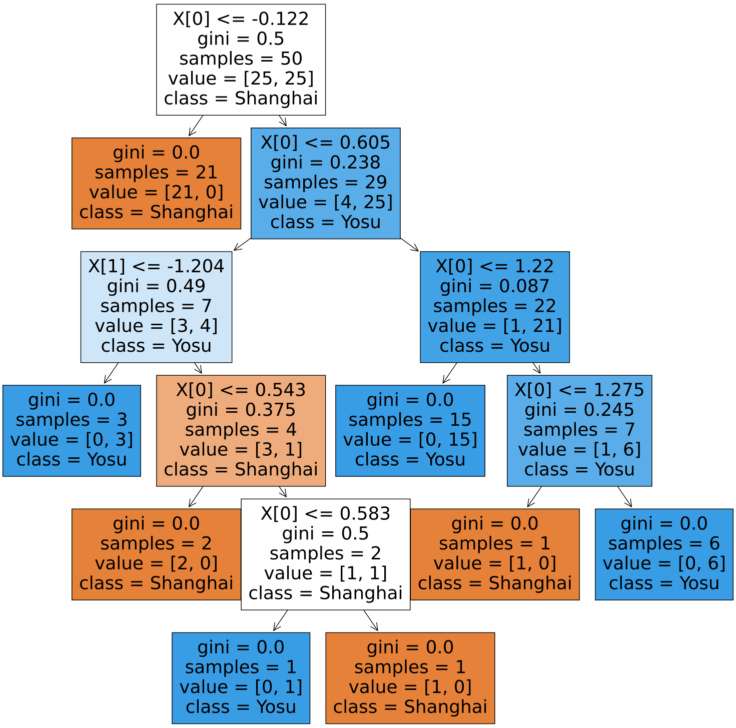


Figure 3.6: Example of a binary classification problem of predicting ports Shanghai or Yosu using a decision tree, where X[0] and X[1] are mock features.

Gini impurity is a measure utilized by tree-based algorithms to decide the most suitable feature to split at each node. It quantifies the impurity or the degree of uncertainty at a node. The Gini impurity is calculated as:

$$Gini(pt) = 1 - \sum (pt_i)^2, \tag{3.8}$$

where pt refers to the probability distribution of the classes at a given node in the decision tree, and pt_i denotes the probability of an element being classified as class i. The Gini impurity ranges between 0 and 1, where 0 indicates that all samples at a node belong to a single class, signifying no uncertainty or impurity. Conversely, a Gini impurity of 1 indicates that the elements are randomly distributed across various classes, signifying maximum impurity or uncertainty.

In decision trees, at each node, the algorithm selects the feature that reduces the Gini impurity the most, thus decreasing uncertainty about the target variable. This process continues recursively until the tree reaches a maximum depth or other stopping criteria are met.

Once the decision tree is built, new data can be classified by traversing the tree from the root node to a leaf node based on the values of the input features. The prediction is then the class label associated with the leaf node.

3.5.2 One-vs-Rest classification

One-vs-rest (OvR), also known as one-vs-all, is a heuristic method used for multiclass classification problems (Rifkin and Klautau 2004). The fundamental idea of OvR is to decompose the multiclass classification problem into multiple binary classification problems. Each binary classification problem is solved independently, and the final class prediction is made based on the outputs of all binary classifiers.

Given a multiclass classification problem with K classes, in the OvR approach, we construct K different binary classifiers, each responsible for distinguishing one class from all the remaining classes. In the context of predicting vessel destinations and discharge destinations, this implies that for each destination port $k \in \mathcal{Y}$, we train a binary classifier f_k that classifies each voyage as either going to destination k or not.

Formally, let $f_k : \mathbb{R}^p \to \{0,1\}$ be the binary classifier for class k, trained to predict whether a voyage goes to destination k or not. The classifier f_k is trained on a modified version of the original dataset, where the voyages going to destination k are considered as positive instances and all other voyages are considered as negative instances. In terms of the response vector, we create a modified response vector $\boldsymbol{y}^{(k)}$ where $y_i^{(k)} = 1$ if $y_i = k$ and $y_i^{(k)} = 0$ otherwise. Then we train f_k to predict $\boldsymbol{y}^{(k)}$ based on $\mathbf{X}$:

$$f_k(\mathbf{x}_i) = \begin{cases} 1, & \text{if } y_i = k, \\ 0, & \text{otherwise.} \end{cases} \tag{3.9}$$

In the prediction phase, each binary classifier gives a prediction for a new observation $\mathbf{x}$. The observation is then assigned to the class with the highest predicted probability. Mathematically, the final prediction $\hat{y}$ for a new observation $\mathbf{x}$ is given by

$$\hat{y} = \arg\max_{k \in \mathcal{Y}} f_k(\mathbf{x}). \tag{3.10}$$

In this thesis, we employ Extreme gradient boosting as the base estimator for each binary classifier, which is explained in the next section.b

3.5.3 Extreme gradient boosting

Extreme Gradient Boosting, or XGBoost, is a scalable and efficient implementation of the gradient boosting framework, which can be used for both regression and classification problems (T. Chen and Guestrin 2016). This method is particularly suitable for our problem due to its ability to to model non-linear relationships, given the complex interplay of factors influencing the decision-making process for vessel destination. The robustness of XGBoost to outliers also stands to be advantageous, considering the unpredictable nature of the shipping industry often leads to data anomalies.

XGBoost uses an additive strategy; it builds an ensemble of weak learners, typically decision trees, to create a robust model. In each iteration, a new decision tree is added to the model that minimizes the objective function.

The objective function in XGBoost consists of a loss function and a regularization term. The loss function measures how well the model fits the data, and the regularization term controls the complexity of the model to prevent overfitting.

Given a feature matrix $\mathbf{X} \in \mathbb{R}^{n \times p}$ and a response vector $\boldsymbol{y} \in \mathcal{Y}^n$ where $\mathcal{Y} = \{1, 2, \ldots, K\}$, XGBoost constructs an ensemble of T decision trees. XGBoost then employs the softmax objective function, which is a generalized logistic loss function for multiple classes. Each tree f_t outputs a K-dimensional probability vector, where each element corresponds to a score for one class. The final class prediction is then obtained by applying the softmax function to these scores:

$$\hat{y}_i = \operatorname{argmax}_k \left(\frac{\exp\left(\sum_{t=1}^{T} f_{t,k}(\mathbf{x}_i)\right)}{\sum_{k'=1}^{K} \exp\left(\sum_{t=1}^{T} f_{t,k'}(\mathbf{x}_i)\right)} \right),$$

where $f_{t,k}$ denotes the score given by tree t for class k. XGBoost learns the trees in a sequential manner. For each tree, it tries to minimize a regularized objective function, involving the log loss function.

The log loss function involves the predicted probabilities, rather than the final class predictions. While $\hat{\boldsymbol{y}}$ represents the predicted response vector where $\hat{y}_i$ corresponds to the predicted destination class for the i-th vessel voyage, we should clarify that within the context of the log loss, we use $\hat{p}_{i,k}$, which represents the predicted probability that observation i belongs to class k. This is a component of the predicted probabilities that inform $\hat{\boldsymbol{y}}$. Thus, the multiclass log loss function is given by:

$$l(\boldsymbol{y}, \hat{\boldsymbol{y}}) = -\sum_{i=1}^{n}\sum_{k=1}^{K} \mathbb{1}(y_i = k)\log(\hat{p}_{i,k}), \tag{3.11}$$

where:

- $\mathbb{1}(y_i = k)$ is an indicator function that equals 1 if the true class of observation i is k, and 0 otherwise.
- $\hat{p}_{i,k}$ is the predicted probability that the i-th vessel voyage goes to the k-th destination port.

The regularization term Ω used by XGBoost is defined as:

$$\Omega(f) = \gamma Tl + \frac{1}{2}\lambda||w||^2, \tag{3.12}$$

where f represents a specific tree model in the boosting process, Tl is the number of leaves in the tree, w is the vector of scores on the leaves, and γ and λ are hyperparameters that control the complexity of the model.

The overall objective function is given by:

$$Obj(\Theta) = L(\boldsymbol{y}, \hat{\boldsymbol{y}}) + \Omega(f), \tag{3.13}$$

where Θ represents the parameters of the model, such as the amount of boosting rounds and the maximum depth of the trees.

In each iteration, XGBoost adds a new tree that minimizes this objective function. This is achieved by a second-order Taylor expansion of the loss function, which enables the algorithm to capture more complex relationships in the data. The prediction of the model is updated iteratively as follows:

$$\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + \eta \cdot f_t(\mathbf{x}_i), \tag{3.14}$$

where $\hat{y}_i^{(t)}$ is the prediction of the i-th instance at the t-th iteration, $f_t(\mathbf{x}_i)$ is the prediction of the t-th tree, and η is the learning rate, which often needs to be tuned.

The final prediction for the i-th instance is then the class with the highest probability:

$$\hat{y}_i = \arg\max_{k \in \mathcal{Y}} p(y_i = k|\mathbf{x}_i). \tag{3.15}$$

XGBoost has a number of hyperparameters that control the size and complexity of the ensemble, including the learning rate, maximum depth of the trees, number of trees, and regularization parameters. The optimal values of these hyperparameters need to be determined through a process of hyperparameter tuning, which will be discussed in the next section.

3.5.4 Hyperparameter optimization

In the optimization process of tuning the hyperparameters of an XGBoost model, the Tree-structured Parzen Estimator (TPE) algorithm provided by the Hyperopt library can be used. TPE is a Bayesian optimization algorithm that is particularly effective in optimizing complex, high-dimensional functions with relatively few function evaluations (Bergstra et al. 2011). The heart of the TPE algorithm lies in the construction and adaptation of two conditional probability density functions (PDFs), $l(\theta)$ and $g(\theta)$, which correspond to the likelihood of a hyperparameter configuration θ given the prior evaluation results ψ.

An important concept here is the introduction of a quantile threshold, denoted $\hat{\psi}$. This threshold is employed to discern between 'good' and 'bad' configurations of the hyperparameters. If a particular configuration results in performance ψ that is better than $\hat{\psi}$, it is considered 'good', and is captured by the PDF $l(\theta) = p(\theta \mid \psi < \hat{\psi})$. In contrast, configurations that yield performance ψ that is equal to or worse than $\hat{\psi}$ are considered 'bad', and are represented by the PDF $g(\theta) = p(\theta \mid \psi \geq \hat{\psi})$.

In the optimization step, the TPE algorithm seeks to maximize the Expected Improvement (EI) criterion, which is the expectation of improvement over the current best solution, with respect to the hyperparameter configuration θ. The EI criterion in the TPE algorithm is somewhat complex, as it depends on the increase in the ratio $\frac{l(\theta)}{g(\theta)}$.

Given a current best solution θ^* (which maximizes the ratio $\frac{l(\theta)}{g(\theta)}$ among all evaluations so far), the EI(θ) at a new point θ is derived from the following:

$$EI_{\psi^*}(\theta) \propto \left(\gamma + \frac{g(\theta)}{l(\theta)}(1-\gamma)\right)^{-1}, \tag{3.16}$$

where $\gamma = p(\psi < \psi^*)$. This expression shows that to maximize improvement, we would like points θ with high probability under $l(\theta)$ and low probability under $g(\theta)$.

In practice, the TPE algorithm does not explicitly calculate $EI_{\psi^*}(\theta)$. Instead, it selects the next point θ to evaluate by maximizing the ratio $\frac{l(\theta)}{g(\theta)}$. The $EI_{\psi^*}(\theta)$ criterion is implicit in this selection process, as the point θ that maximizes the ratio is also expected to provide the greatest improvement. Thus, the hyperparameter configuration θ that yields the highest ratio is chosen for the next evaluation, $\theta^* = \arg\max_\theta \frac{l(\theta)}{g(\theta)}$.

As new evaluations are made, the models $l(\theta)$ and $g(\theta)$ are updated, hence incorporating the most recent information to guide the next selection. The process continues until a predetermined stopping criterion is met, such as when a maximum number of function evaluations is reached or the performance improvement between iterations falls below a defined minimum.

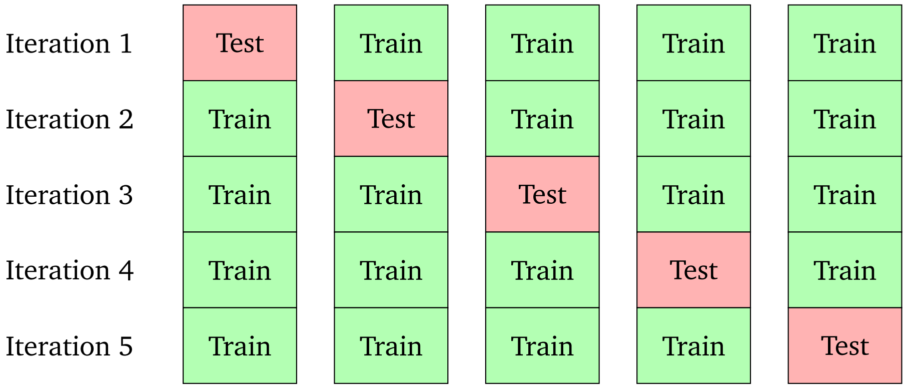
By this means, the TPE algorithm leverages the principles of Bayesian inference and statistical modelling to balance the trade-off between exploration and exploitation in the hyperparameter space, while mitigating computational expense.

3.5.5 Cross-validation

Cross-validation is a resampling procedure used to evaluate machine learning models on a limited data sample (Stone 1978). The procedure has a single parameter called g that refers to the number of groups that a given data sample is to be split into. As such, the procedure is called g-fold cross-validation. Here are the steps involved in g-fold cross-validation:

1. Shuffle the dataset randomly.
2. Split the dataset into g groups or folds.
3. For each unique group:
    - Take the group as a hold out or test data set.
    - Take the remaining groups as a training data set.
    - Fit a model on the training set and evaluate it on the test set.
    - Retain the evaluation score and discard the model.
4. The result of g-fold cross-validation is often given as the mean of the model skill scores.

Here is a basic figure representing 5-fold cross-validation:



Iteration 1	Test	Train	Train	Train	Train
Iteration 2	Train	Test	Train	Train	Train
Iteration 3	Train	Train	Test	Train	Train
Iteration 4	Train	Train	Train	Test	Train
Iteration 5	Train	Train	Train	Train	Test

Here, the data set is split into 5 folds. In each iteration, one fold is used for testing and the remaining folds are used for training. This process is repeated until each fold has been used as the test set once. The splitting in this thesis is done using a stratified train-test split. This is a technique for partitioning data that is especially useful when dealing with imbalanced datasets. The stratification process ensures that each subset of the data is representative of all strata of the data. In this context, a stratum refers to a category or class in the target variable.

In other words, stratified train-test split ensures that each class (or stratum) is proportionally represented in both the training and test sets. For instance, consider a binary classification problem where the target variable is imbalanced: 80% of instances belong to class 0, and only 20% belong to class 1. A random train-test split might create a training set with few instances of the minority class (class 1), causing the machine learning model to be biased towards predicting the majority class (class 0).

However, if a stratified train-test split is used, it maintains the original 80%-20% distribution of classes in both the training set and the test set. This leads to a more representative subset of data for both training and evaluating the model, and is particularly important for this problem due to the large imbalance in arrival port distribution.

Stratified g-Fold cross-validation

In classification problems, particularly those with severe class imbalance such as this one, preserving the original class distribution in each training and test set is critical. A commonly used method that adheres to this principle is Stratified g-Fold Cross-Validation. Stratified g-Fold Cross-Validation is a variant of traditional g-Fold Cross-Validation, as explained in the methodology, that stratifies the data, ensuring that each fold is a good representative of the overall class distribution of the data.

However, a single run of Stratified g-Fold Cross-Validation might yield a noisy estimate of the model's performance. This noise is attributable to the variability in the data splits, where different partitions can lead to substantially different results. To alleviate this issue, we introduce Repeated Stratified g-Fold Cross-Validation.

In Repeated Stratified g-Fold Cross-Validation, the Cross-Validation procedure is repeated multiple times, with different random splits into K folds each time. The performance measure reported by this method is then an average of the values computed in the loop. This average is expected to be a more robust and accurate estimate of the model's performance.

$$\text{Performance}_{\text{RepeatedStratifiedgFold}} = \frac{1}{N}\sum_{i=1}^{N}\text{Performance}_{\text{StratifiedgFold},i}, \tag{3.17}$$

where N is the number of repeats, and $\text{Performance}_{\text{StratifiedgFold},i}$ is the performance measure of the i-th Stratified g-Fold Cross-Validation run.

The major advantage of Repeated Stratified g-Fold Cross-Validation is that it provides a more reliable estimate of model performance. However, this comes at the cost of computational efficiency, as we need to fit and evaluate many more models. Specifically, if we use R repeats of g-fold cross-validation, we would need to fit and evaluate $R \times g$ models. This cost can be mitigated if the computations are distributed across multiple cores or machines, speeding up the process significantly, which can be done for bagging algorithms with great success.

3.6 Evaluation metrics

In this section, we will discuss the top-κ accuracy metrics, Port frequency-based decision strategy(PFD), the average prediction distance error (APDE), median prediction distance error (MPDE), and the Multiclass Brier score as evaluation measures.

3.6.1 Top-κ Accuracy

In multiclass classification problems, each class is assigned a probability, where the class with the highest probability is normally selected as the prediction. Denote the estimated probability matrix as $\mathbf{Q} \in [0,1]^{n \times K}$, where each row $\boldsymbol{q}_i$ represents the probability distribution over the K classes for the i-th vessel voyage. Each element q_{ij} represents the predicted probability that the i-th vessel voyage destination belongs to the j-th class.

Top-κ accuracy is a generalization of this traditional accuracy metric. Instead of considering only the top 1 prediction, it takes into account the top-κ predictions, the classes with top-κ probabilities, made by the model. The top-κ accuracy measures the proportion of instances for which the correct class label appears in the top-κ predictions.

Let Z be the total number of predictions, κ be the number of predictions to consider, y_i be the true class label for the i-th instance, and $\hat{y}_{i,j}$ be the j-th predicted class label for the i-th instance. Then, the top-κ accuracy (Acc_κ) can be defined as:

$$\mathrm{Acc}_\kappa = \frac{1}{Z}\sum_{i=1}^{Z}\mathbb{I}\big(y_i \in \{\hat{y}_{i,1}, \hat{y}_{i,2}, \ldots, \hat{y}_{i,\kappa}\}\big)$$

where $\mathbb{I}\big(y_i \in \{\hat{y}_{i,1}, \hat{y}_{i,2}, \ldots, \hat{y}_{i,k}\}\big)$ is an indicator function that evaluates to 1 if the true label y_i is among the top-κ predictions and 0 otherwise. Note that by setting $\kappa = 1$ one recovers the global accuracy.

3.6.2 Port frequency-based decision strategy

In multiclass classification problems, it is often important to consider the impact of different factors on the decision-making process of the maximum probability. One such factor is the frequency of departure ports. The Port Frequency-based Decision(PFD) strategy aims to incorporate the departure port frequencies into the classification model to make more informed decisions (Zhang et al. 2020). This is also a method to mitigate eventual overfitting, since the strategy is a means of reducing bias towards certain ports that may be over-represented in the data. To use this strategy, the final decision making is not based solely on the highest probability in the probability vector obtained from the classifier, but a port frequency-based normalized probability vector.

We compute the frequency vector $\mathbf{f} = (f_1, f_2, ..., f_K)$, where each element f_j represents the frequency of departures for the j-th class.

Now, we perform the port frequency-based normalization process which adjusts the probability matrix $\mathbf{Q}$ based on the departure frequencies. This can be expressed as:

$$q'_{ij} = \frac{q_{ij}}{f_j}, \tag{3.18}$$

for $i = 1, 2, ..., n$ and $j = 1, 2, ..., K$. This operation normalizes each probability by its respective class frequency.

The resulting matrix $\mathbf{Q}'$ does not necessarily maintain the row-wise sum equals to 1 anymore. To get a valid probability distribution, a further normalization step is necessary:

$$q''_{ij} = \frac{q'_{ij}}{\sum_{j=1}^{K} q'_{ij}}, \tag{3.19}$$

for $i = 1, 2, ..., n$ and $j = 1, 2, ..., K$. This ensures that each row in the final probability matrix $\mathbf{Q}''$ sums up to 1, thus representing valid probability distributions for each vessel voyage's destination. The new prediction is then the port with the highest probability in each corresponding row in $\mathbf{Q}''$.

This PFD strategy ensures that the class predictions are not skewed towards over-represented classes by taking into account the departure frequency of each class.

3.6.3 Average prediction distance error

The average prediction distance error (APDE) provides a measure of the average distance between the predicted and true class labels. It quantifies the average discrepancy between the predicted and actual port for each instance. Let M be the total number of incorrect predictions in a test set, Ξ_i be the latitude and longitude of the true class label y_i for the i-th instance, and $\hat{\Xi}_i$ be the latitude and longitude of the the predicted class label $\hat{y}_i$. Then, the APDE can be defined as:

$$\mathrm{APDE} = \frac{1}{M}\sum_{i=1}^{M} d_{\text{H-sine}}(\Xi_i, \hat{\Xi}_i),$$

where $d_{\text{H-sine}}$ is the Haversine distance as defined in Equation (2.6), in kilometers.

3.6.4 Median prediction distance error

The median prediction distance error (MPDE) represents the central tendency of the prediction distance errors and provides a robust measure against outliers in the error distances. It is the median value of the prediction error distances across all predictions.

Let M be the total number of incorrect predictions, Ξ_i be the latitude and longitude of the true class label y_i for the i-th instance, and $\hat{\Xi}_i$ be the latitude and longitude of the predicted class label $\hat{y}_i$. The MPDE can be defined as the median of the set of prediction error distances

$$\mathrm{MPDE} = \mathrm{median}\left(\left\{d_{\text{H-sine}}(\Xi_i, \hat{\Xi}_i)\right\}_{i=1}^{N}\right).$$

3.6.5 Brier score

The Brier Score (Brier et al. 1950) is an effective metric for assessing the accuracy of probabilistic predictions in multiclass classification problems, and has been used previously with assessing XGBoost (Jullum et al. 2020). In contrast to other metrics such as accuracy or log loss, the Brier Score has a clear probabilistic interpretation and is particularly informative when it comes to understanding model performance across multiple classes.

The Brier Score measures the mean squared difference between the predicted probabilities and the actual outcomes for each class. This provides a comprehensive view of the model's performance, as it penalizes both type I and type II errors and takes into account the distance between the predicted probabilities and the actual outcomes. As a result, the Brier Score provides a good balance between precision and recall.

Additionally, the Brier Score promotes calibrated predictions. This means the model is encouraged to make predictions that align with the true class proportions observed in the data. In the context of predicting vessel destinations, this property of the Brier Score is particularly beneficial, since the accuracy of the predictions are important. The Brier Score is bounded between 0 and 1, with a score of 0 indicating perfect accuracy and a score of 1 indicating complete disagreement between the predicted probabilities and the actual outcomes. This range facilitates an intuitive understanding of model performance.

Again, denote the estimated probability matrix as $\mathbf{Q} \in [0,1]^{n \times K}$. We can also express the true labels $\boldsymbol{y}$ in a one-hot encoded matrix form $\boldsymbol{Y} \in \{0,1\}^{n \times K}$, where $Y_{ij} = 1$ if the i-th vessel voyage destination belongs to the j-th class, and $Y_{ij} = 0$ otherwise.

The multiclass Brier score is then defined as:

$$BS = \frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{K} (q_{ij} - Y_{ij})^2 \tag{3.20}$$

3.7 Feature importance

When predicting dry bulk vessel destinations, geographical features and ship-specific details play an integral role. For example, the departure port and its coordinates provide information about the origin of a voyage, which is likely to influence the destination due to factors such as trade routes, shipping lanes, and bilateral trade agreements. Other features derived from vessel trajectories, like the vessel's speed, heading, or even past voyage patterns, can also be significant in predicting the destination. These trajectory-derived features can help identify common patterns or trends in routes which can, in turn, improve the prediction accuracy of the model.

On the other hand, ship-specific details such as the owner, vessel type, size, and age could provide additional insights. Different owners might have different preferences or specializations in terms of trade routes or cargo types. Similarly, the vessel's type and size can influence its possible destinations as it determines the kind of ports it can dock at and the type of cargo it can carry.

The interplay between these various features can create a complex predictive problem. Understanding the importance of these features is not just about improving model performance; it can also provide strategic insights for various stakeholders, such as the shipbrokers, in the shipping industry. By identifying the features that have the most influence on vessel destinations, industry participants can make more informed decisions and develop better strategies for operations and planning. Two different types of feature importance will be introduced.

3.7.1 Permutation feature importance

Permutation Feature Importance (PFI) is a method utilized for understanding the importance of different features (variables) in a machine learning model's predictive power (Altmann et al. 2010). The core idea behind PFI is to measure the decrease in a model's performance when one feature's values are randomly shuffled, thereby breaking the relationship between the feature and the true outcome.

To compute the PFI of a particular feature x, we perform the following steps:

1. Train a machine learning model using the full dataset and compute its baseline performance Pf_{baseline}. This could be any appropriate metric such as accuracy

for classification problems or R^2 for regression problems. In this thesis, the Top-1 accuracy is used.

2. Permute the values of feature x in the test set. This breaks the association between feature x and the true outcome.
3. Use the model to make predictions on this permuted dataset and compute the permuted performance Pf_{permuted}.
4. The PFI of feature x is then given by the decrease in performance caused by the permutation:

$$PFI_f = Pf_{\text{baseline}} - Pf_{\text{permuted}}. \tag{3.21}$$

The permutation process can be repeated n times for more robust results, with the final PFI calculated as the average of these repetitions. This way, the resulting PFI provides a ranking of feature importance, with higher values indicating a higher importance in the predictive model. Since the process is repeated n times, there is some deviation to the reduction in score, which can be displayed using e.g boxplots.

3.7.2 Shapley values in machine learning

Shapley values, originating from cooperative game theory, have gained significant attention in recent years as a powerful method for interpreting and explaining the predictions of machine learning models (Shapley 1997). In this section, we provide an overview of Shapley values and discuss their application in this thesis for understanding the contribution of individual features in predicting vessel destinations.

The concept of Shapley values was introduced to fairly allocate the payoff among players in cooperative games, considering each player's contribution to the total payoff. The Shapley value for a player can be interpreted as the average marginal contribution of the player to all possible coalitions.

Given a cooperative game with n players and a characteristic function $v : 2^N \rightarrow \mathbb{R}$ that assigns a real value to every coalition of players, the Shapley value $\zeta_i(v)$ for player i can be computed as:

$$\zeta_i(v) = \sum_{S \subseteq P \setminus \{i\}} \frac{|S|!(p - |S| - 1)!}{p!} [v(S \cup \{i\}) - v(S)], \tag{3.22}$$

where $P = \{1, 2, \ldots, p\}$ is the set of players, and S is a subset of players not including player i.

In the context of machine learning and multiclass classification problems (Lundberg and S.-I. Lee 2017), Shapley values can be used to explain the output of a model by fairly attributing the prediction for each class to each input feature. The idea is to treat each feature as a player in a cooperative game, and the prediction for each class is considered as the payoff. The Shapley value for a feature indicates its average contribution to the prediction across all possible combinations of features and all classes.

Given a machine learning model $f : \mathbb{R}^{n \times p} \to \mathcal{Y}^n$, the Shapley value $\phi_{j,k}(f, \mathbf{x}_i)$ for the j-th feature of the i-th instance, contributing to the k-th class, can be computed using a similar formula as in cooperative games:

$$\zeta_{j,k}(f, \mathbf{x}_i) = \sum_{S \subseteq P \setminus \{j\}} \frac{|S|!(p - |S| - 1)!}{p!} [f_k(\mathbf{x}_{i, S \cup \{j\}}) - f_k(\mathbf{x}_{i,S})], \tag{3.23}$$

where $\zeta_{j,k}(f, \mathbf{x}_i)$ is the Shapley value for the j-th feature contributing to the k-th class of the i-th instance. It represents the average contribution of the j-th feature across all possible subsets of features, considering each possible subset size.

The summation iterates over all subsets S of the feature set P_x excluding the feature j. For each subset, the term $[f_k(\mathbf{x}_{i,S \cup j}) - f_k(\mathbf{x}_{i,S})]$ calculates the difference in the model's output for the k-th class when the j-th feature is included and when it is excluded.

The term $\frac{|S|!(p-|S|-1)!}{p!}$ is a weighting factor that represents the number of ways the particular subset S can be formed, and it normalizes the total contribution of the j-th feature across all subsets.

The vectors $\mathbf{x}_{i,S\cup j}$ and $\mathbf{x}_{i,S}$ represent the i-th instance where the features in the set $S \cup j$ and S are active, respectively, while the remaining features are replaced by their expected values. This acts as a way of simulating the impact of including or excluding a specific feature from the model.

In essence, Shapley values provide a powerful and fair way to attribute the prediction of a machine learning model to its features, considering all possible combinations of features and offering insights that simple feature importance metrics can't provide. They enable us to capture the complex interactions between features and their impact on model predictions, which is crucial in understanding and explaining complex machine learning models.

Interpreting SHAP summary plots

SHAP summary plots offer a comprehensive visualization of the feature importances and their impact on the model's predictions, and are commonly used in Explainable AI (Xu et al. 2019). An example plot from the SHAP website is provided in Figure 3.7, where 8 different features for an example problem are displayed, showing a varied distribution of SHAP values.

They provide an overview of the Shapley values for each feature across all instances in the dataset. Here is a brief guide on how to interpret these plots:

- **Features:** Each row in the plot represents a feature in the dataset. Features are usually sorted by their average absolute Shapley values, with the most important feature on top and the least important at the bottom.
- **Shapley Values:** The plot displays the Shapley values for each feature-instance pair as individual data points. The horizontal position of a data point indic-

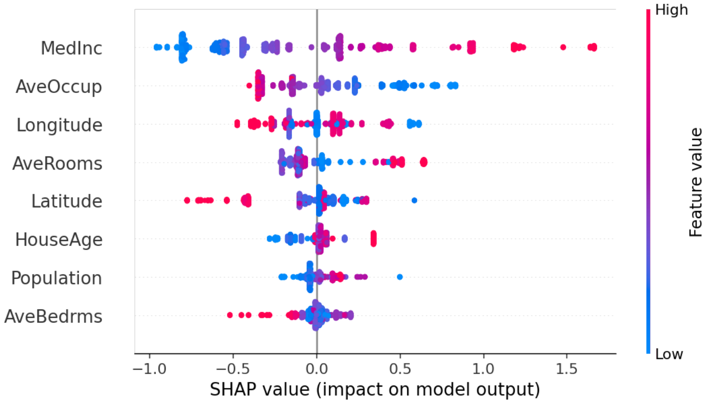


Figure 3.7: Example plot of SHAP values (Lundberg, 2008)

ates the magnitude and direction of the Shapley value for that specific instance. Positive Shapley values (to the right of the vertical axis) indicate that the feature contributes to increasing the prediction, while negative values (to the left) suggest that the feature decreases the prediction.

- **Color:** Data points are colored according to the value of the corresponding feature. Typically, a gradient color scheme is used, with one color representing low feature values and another color representing high feature values. This coloring scheme helps in identifying trends and interactions between features and their values.
- **Feature Importance:** The overall importance of a feature can be estimated by the spread and magnitude of its Shapley values in the plot. Features with larger spreads and higher average absolute Shapley values are generally more important for the model's predictions.
- **Feature Effects:** By examining the distribution of Shapley values and their corresponding feature values (indicated by colors), one can gain insights into how the feature values influence the model's predictions. For example, a positive correlation between the Shapley values and feature values suggests that higher feature values lead to higher predictions, and vice versa.

For instance, in Figure 3.7, one can see that the most important feature is MedInc, while the least importance feature is AveBedrms. For MedInc, low SHAP values of the feature negatively contribute to the prediction, while larger values contribute positively. Since the spread is quite large for MedInc, this also indicates that it is a more important feature.

Chapter 4

Results and discussion

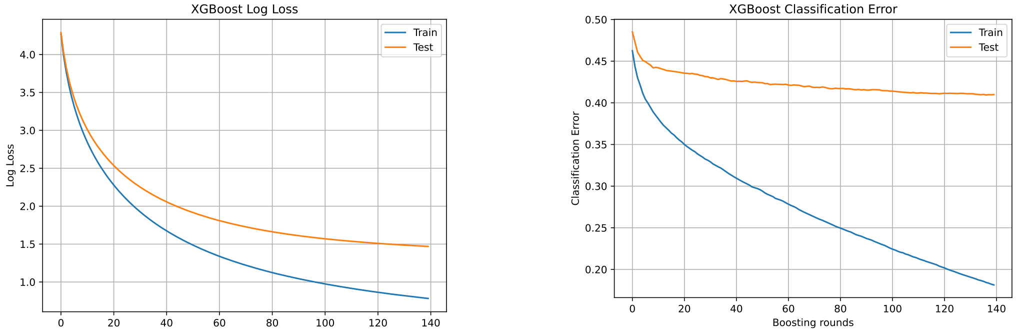
4.1 Overview of the evaluation process

This section explains the evaluation process of our models, focusing on the prediction of destination ports for different purposes, for different vessel sub-segments, VLOC, Capesize, Panamax, and Supramax, for different sets of features. This is first done on the port-to-port dataset, where each voyage is as described in Equation 2.1.

Initially, the MLTD was calculated for different sub-segments. First a test is done comparing Hausdorff versus SSPD, and then the results are displayed for the SSPD, which is the best method for this problem.

Afterwards, we trained three distinct XGBoost models on different datasets for these vessels: one comprised solely of static features, while the second used only trajectory-based features derived from AIS signals, and the third combined both. The purpose was to ascertain the added value of AIS signals in enhancing the model's performance.

The entire process, from model training to evaluation, was done for the VLOC, Capesize, Panamax and Supramax datasets, thereby ensuring our approach's robustness across various vessel sub-segments. The learning curves, with log loss and classification error as metrics, for the Capesize port-to-port predictions are displayed in Figure 4.1, where one can see a clear tendency to overfitting due to the large gap between the training set curve and the validation set curve. Even though XGBoost has many hyperparameters that can be tuned to help mitigate overfitting, this did not help in any significant manner.



(a) Log loss

(b) Classification error

Figure 4.1: Training curves for Port-to-port Capesize using all available features

The destination field from the AIS signals is then added, and the machine learning results with this feature included are discussed. This process was then repeated for the laden and ballast voyages, however only for the Capesize sub-segment.

As previously discussed, VLOCs have a predictable voyage pattern due to their long-term contracts, and they almost exclusively transport goods between Brazil, Australia and China. Therefore, the potential scope of prediction for VLOCs is significantly limited, and further testing for VLOCs in predicting unload and load ports may not provide additional valuable insights. On the other hand, Capesize vessels exhibit a balance between predictability and variability in their voyages, making them an excellent candidate for further testing. While Panamax and Supramax vessels also show promising characteristics for the prediction task, due to their increased route flexibility and variability, it is anticipated that the trends observed for Capesize vessels will hold for these vessel sub-segments as well, where we have seen that Capesize consistently outperforms the other two sub-segments. Therefore the analysis for laden and ballast voyages will be performed only on the Capesize vessels.

4.2 Port-to-port voyages

4.2.1 MLTD for port-to-port voyages

SSPD vs Hausdorff comparison

In the previous chapter, two different similarity measures were introduced. In previous works, it has been generally shown that the SSPD outperforms the Hausdorff distance in similar studies, such as in Besse et al. (2016), therefore we are interested in the accuracy of the MLTD when it is based on either the Hausdorff distance or the SSPD. We denote these two versions of the MLTD classifier as f_H and f_S with $\phi = 10$, as defined in subsection 3.2.3, respectively.

To compare f_H and f_S , we generate n random subsets $S_1, \ldots, S_n$ of the Capesize sub-segment dataset, each of size m (in our case, $n = 100, m = 1000$), and compute their Top-1 accuracies:

$$a_{i,H} = \mathrm{Acc}_1(f_H, S_i), \quad a_{i,S} = \mathrm{Acc}_1(f_S, S_i), \quad i = 1, \ldots, n. \tag{4.1}$$

Finally, we visualize and compare the distributions of $a_{1,H}, \ldots, a_{n,H}$ and $a_{1,S}, \ldots, a_{n,S}$ using box plots, displayed in Figure 4.2. SSPD clearly performs better on average, likely due to that the Hausdorff distance is sensitive to outliers, and the run-time is approximately equal, therefore SSPD is chosen as the preferred similarity measure

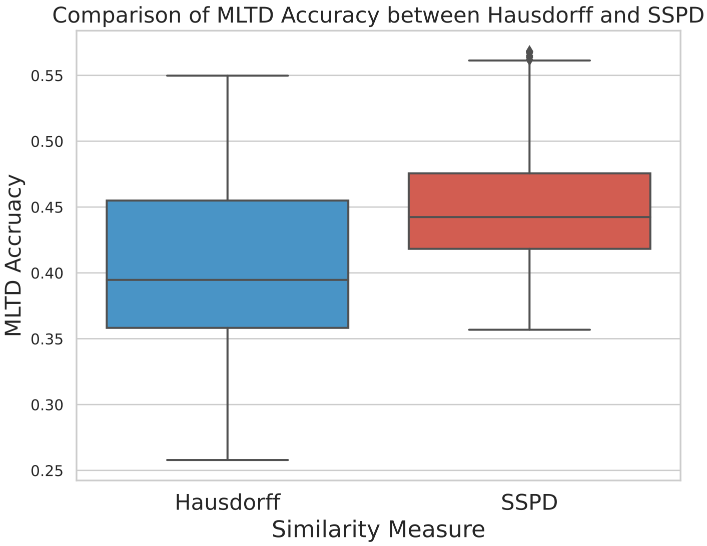


Figure 4.2: Distributions of the Hausdorff and SSPD measures MLTD accuracy distance for all datasets.

Results with SSPD

The MLTD is the predicted port, and MLTDC is the country the predicted port lies in. One can observe distinct differences in the accuracy of predictions based on the sub-segment of the vessel. As seen in Figure 4.3, the accuracy of the port predictions is highest for the VLOC sub-segment, at around 58%. For the Capesize, Panamax and Supramax sub-segments, the accuracies are slightly lower, at around 48%, 41% and 37% respectively. This is in line with theory that the VLOC vessels follow more common routes, and that Capesize follows more predictable routes than Panamax and Supramax.

The accuracy of the country predictions follows a similar pattern, but with notably higher accuracies across all sub-segments. The VLOC sub-segment again performs best with an average accuracy of around 87%, however the Capesize is almost at the exact same accuracy, while the Panamax and Supramax sub-segments show accuracies of 76% and 68%, respectively.

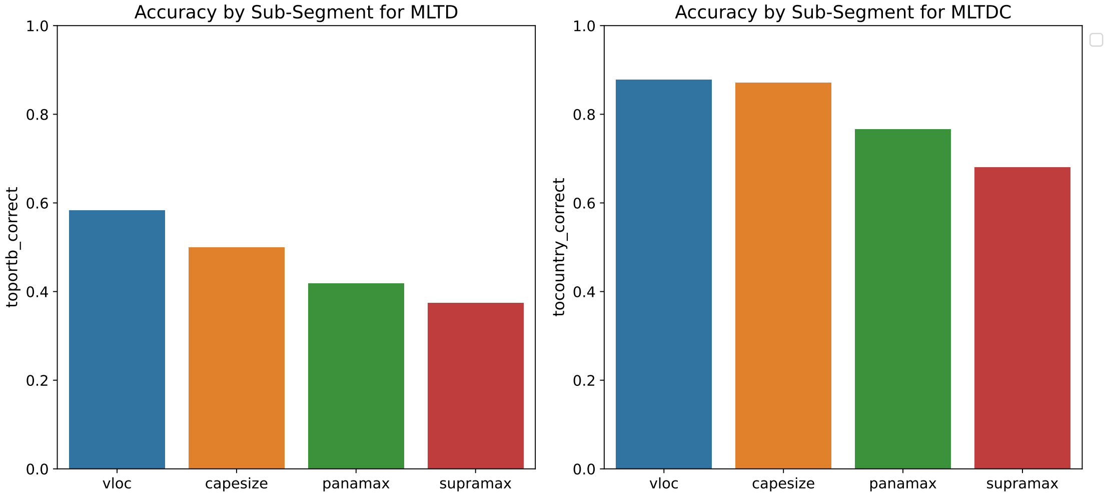


Figure 4.3: Port-to-port MLTD and MLTDC accuracy grouped by sub-segment

When considering the number of days before arrival in Figure 4.4, we find that the prediction accuracy generally decreases as the number of days before arrival increases. This decrease is seen across all sub-segments. As the vessel approaches the destination port, the similarity measure to other trajectories will increasingly align with a smaller subset of trajectories, enhancing its accuracy.

For example, in the Capesize sub-segment, the port prediction accuracy falls from around 56% when the prediction is made one day before arrival, to approximately 42% when the prediction is made seven days before arrival. This downward trend is similarly observed in the Panamax and Supramax sub-segments.

The country prediction accuracies show a similar pattern. In the Capesize sub-segment, the accuracy decreases from 90% one day before arrival, to 80% seven days before arrival. The other sub-segments also follow this trend, with the Panamax sub-segment

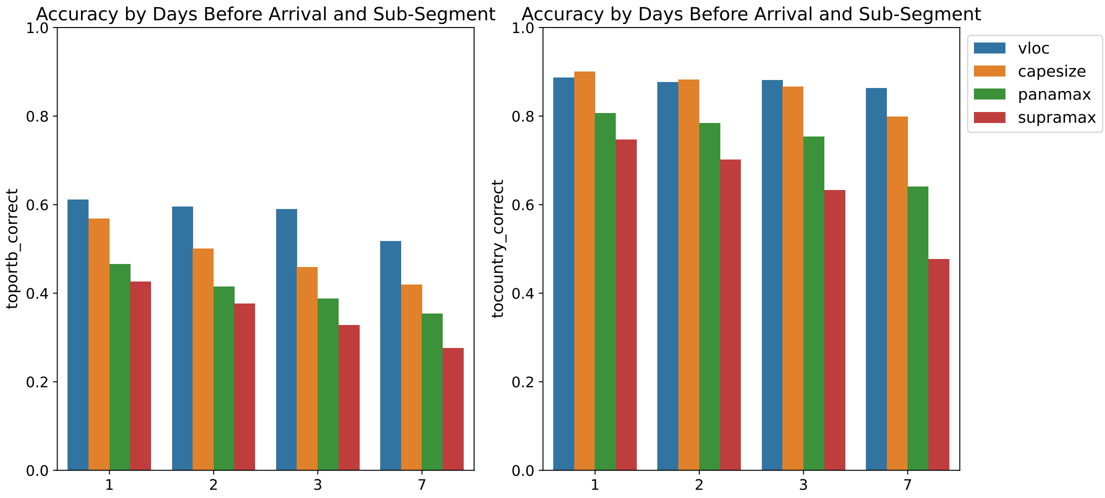


Figure 4.4: MLTD accuracy grouped by days and sub-segment

showing a decrease from 80% to 64%, and the Supramax sub-segment from 74% to 47%.

This analysis suggests that the MLTD accuracy is more reliable for shorter time horizons, and for vessels in the VLOC and Capesize sub-segments. As MLTD is solely a spatial attribute, it will be utilized as a feature in machine learning, alongside other relevant variables such as departure port and vessel information, which might increase the accuracy.

The geographical aspect is also interesting to analyze. A normalized confusion matrix showing the recall for the top 30 busiest arrival ports is displayed in Figure 4.5. The confusion matrix presents the recall, showing clear struggles with ports that are close to each other, such as ports in China, Huanghua, Tianjin, Qingdao, Jingtang, Rizhao and Caofeidan and a few more, and as well a clear confusion between Port Hedland, Port Walcott and Dampier, where Port Hedland is often incorrectly predicted.

To see if this is a general problem, the accuracy for each departure port is displayed on a world map in Figure 4.6, where each possible destination port is plotted. The

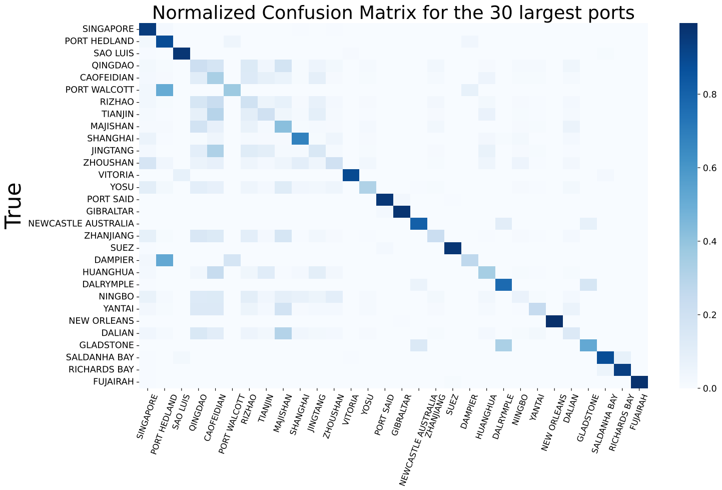


Figure 4.5: Confusion matrix of the top 30 busiest arrival ports, normalized to present recall

ports are scaled in regards to accuracy, as well as color coded where green indicates a higher accuracy and blue indicates a lower. One can see a general trend that around the larger, green circles, there are more smaller blue circles, indicating that the MLTD overfits to some larger ports. This especially holds true for the Chinese coast.

4.2.2 Model selection

For destination port prediction, several state-of-the-art machine learning models, with default hyperparameters, are tested on a particular configuration of the dataset, for Capesize vessels with all features. It is assumed due to the nature of the dataset, that this will hold across configurations.

A repeated stratified 5-fold cross validation is used to ensure robustness and comparability, and the mean Top-1 Accuracy is used as a metric for the performance. This is

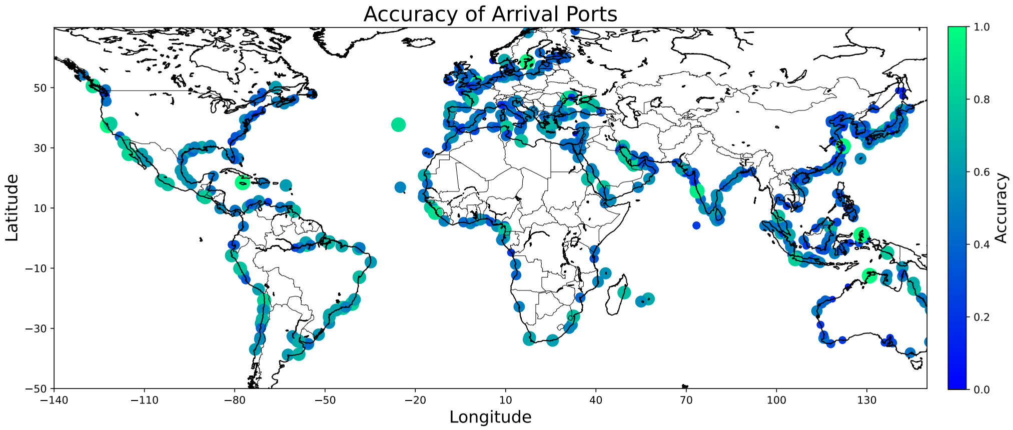


Figure 4.6: MLTD Accuracy of each arrival port

done on a subset of 10000 samples of Capesize vessels.

Table 4.1: Model selection using repeated stratified 5-fold cross validation

Model	Mean Top-1 Accuracy(%)
Random Forest	51.30
Gaussian Naive Bayes	38.72
KNN	10.80
XGBoost	56.04
Extra Trees	50.34
One vs Rest(XGBoost)	55.52

The results of the model selection are displayed in Table 4.1, where one can see that XGBoost outperforms the other models. The only model that is close is One vs rest multiclass classification with XGBoost as a base classifier. however the run-time is much larger than for regular XGBoost, since K models need to be trained, therefore regular XGBoost is preferred. The run-time for XGBoost was also longer than for the others, considering the creation of trees in e.g. Random Forests which is inherently parallel, however the added accuracy was preferred ahead of run-time. Each following model also uses the TPE algorithm as explained in the previous chapter for hyperparameter tuning, to ensure a good fit of the hyperparameters. For the XGBoost above, training the model with tuned hyperparameters increased the accuracy from 56.04 % to 60.04 %.

4.2.3 Machine learning for port-to-port voyages

Initially, when the model was trained with only static features as shown in Table 4.2, the Top 1 accuracy varied between 26.9 % for Supramax and 73.4 % for VLOC, with the Brier Score spanning from 0.36 to 0.88 across all vessel sub-segments. Moreover, the implementation of only trajectory features resulted in considerable enhancement of model's performance. As per Table 4.3, the model achieved a Top 1 accuracy ranging from 48.6 % for Supramax to 64.8 % for VLOC, and reduced the Brier Score range to 0.45 to 0.65.

Table 4.2: Prediction Results for Port-to-Port with only static features

Segment	**Accuracy** (%)					**Error**		
	Top 1	**Top 3**	**PFD**	**Cluster**	**Country**	**APDE(km)**	**MPDE(km)**	**Brier Score**
VLOC	73.4	93.6	73.4	78.1	90.2	2046.8	872.6	0.36
Capesize	36.8	58.9	37.1	44.7	62.4	3792.8	2880.8	0.77
Panamax	31.8	51.3	32.3	37.2	54.1	4615.9	3219.8	0.83
Supramax	26.9	45.1	26.9	29.6	42.2	4257.9	3160.7	0.88

The combined use of static and trajectory features led to the most significant improvement. As presented in Table 4.4, the Top 1 accuracy for Capesize vessels peaked at 59.4 %, an increase over the results when employing each feature set separately. Simultaneously, the Brier Score improved further, falling to as low as 0.45, indicating a better calibrated model. This increase could be seen across all types.

Inspecting the results in greater detail reveals that the VLOC sub-segment consistently

Table 4.3: Prediction Results for Port-to-Port with only trajectory features

Segment	Accuracy (%)					Error		
	Top 1	Top 3	PFD	Cluster	Country	APDE(km)	MPDE(km)	Brier Score
VLOC	64.8	83.9	64.9	75.3	92.1	1164.4	512.2	0.45
Capesize	56.3	76.8	57.3	75.7	87.2	961.7	455.9	0.54
Panamax	52.8	73.0	55.2	63.6	83.7	955.5	485.5	0.59
Supramax	48.6	69.4	65.1	74.6	81.4	1194.7	726.2	0.65

Table 4.4: Prediction Results for Port-to-Port with static and trajectory features

Segment	Days before	Accuracy (%)					Error		
		Top 1	Top 3	PFD	Cluster	Country	APDE(km)	MPDE(km)	Brier Score
VLOC	All days	75.3	92.1	84.03	84.00	96.0	1037.2	473.8	0.31
	1	78.5	93.9	-	87.9	96.0	1131.5	395.1	0.28
	2	76.5	92.8	-	84.6	95.9	951.6	473.8	0.29
	3	74.8	91.5	-	82.9	96.5	905.5	473.8	0.32
	7	69.9	89.3	-	78.9	95.3	1155.1	501.4	0.39
Capesize	All days	59.4	79.8	59.9	69.8	89.5	948.5	447.9	0.51
	1	65.6	82.7	-	77.7	92.3	791.4	383.5	0.45
	2	62.3	81.5	-	71.6	89.6	832.7	440.2	0.49
	3	54.5	78.1	-	64.2	89.6	875.6	447.9	0.56
	7	49.0	73.2	-	58.3	82.5	1323.8	717.2	0.64
Panamax	All days	55.0	75.7	56.9	65.4	84.8	1066.4	528.8	0.57
	1	59.7	79.6	-	73.3	88.8	764.5	383.7	0.52
	2	56.4	77.8	-	67.6	87.1	796.8	404.7	0.56
	3	51.4	72.7	-	61.8	83.4	940.1	545.8	0.61
	7	46.1	66.8	-	47.6	72.4	1764.8	1158.1	0.67
Supramax	All days	49.8	70.9	52.2	56.6	75.3	1225.3	759.1	0.65
	1	55.5	76.6	-	66.1	81.9	837.1	447.9	0.58
	2	50.8	72.8	-	58.3	79.2	1017.2	568.3	0.64
	3	46.8	68.1	-	50.2	71.2	1211.0	847.7	0.68
	7	33.3	51.2	-	36.5	50.5	2323.8	2172.1	0.83

outperforms the Capesize, Panamax and Supramax sub-segments across all scenarios.

The VLOC sub-segment demonstrates a consistently high Top 1 accuracy, for instance, 73.4% with static features alone (Table 4.2), 64.8% with trajectory features (Table 4.3), and peaking at 78.5% when both feature sets are used one day before arrival (Table 4.4). In contrast, even the best-performing Capesize, Panamax, and Supramax models fail to exceed a Top 1 accuracy of 65.6%, 59.7%, and 55.5%, respectively, under similar conditions.

The superior performance of VLOC in port prediction can be attributed to a couple of key factors associated with their operational characteristics. Primarily, VLOC vessels are often engaged in long-term contracts, notably between Australia, Brazil and China (Papadionysiou 2014). These contracts result in highly predictable shipping routes, with fewer port-to-port variations compared to other sub-segments. This consistency enhances the model's ability to accurately predict their port of arrival. Additionally, the VLOC fleet is relatively small, which further reduces variability in port destinations, as fewer vessels result in a narrower range of ports being serviced. Therefore the performance of the Capesize vessels is perhaps more interesting, since they are more volatile.

Excluding VLOC,when both static and trajectory features were used together (as seen in Table 4.4), the Capesize sub-segment again outperformed the others with a maximum Top 1 accuracy of 65.6% and a minimum Brier Score of 0.45.

This consistent superior performance of the Capesize sub-segment compared to Panamax and Supramax, might be attributed to the nature of Capesize vessels. Capesize dry bulk carriers are the largest(excluding VLOC), among dry bulk carriers, and they are often associated with more predictable routes due to their size and the volume of the commodities they transport, such as iron ore and coal. These large cargoes are typically moved in bulk between major ports, which might provide more regular patterns that enhance the prediction capability of the models. The main ports for e.g iron ore lie in China and Australia. AF Code has provided graphs to support this, displayed in Figure 4.7 and 4.8, where one can clearly see that the largest exporter of iron is Australia, and the largest importer is China.

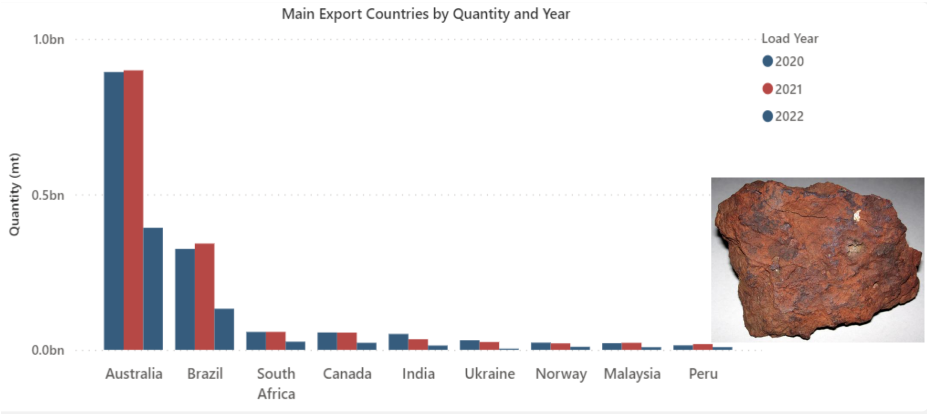


Figure 4.7: Top exporters of iron ore. Source: Fearnleys

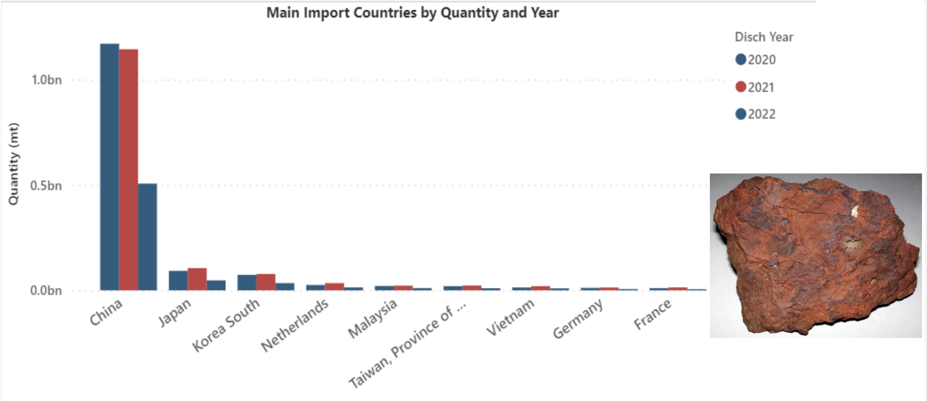


Figure 4.8: Top importers of iron ore. Source: Fearnleys

On the other hand, Panamax and Supramax vessels, being smaller, have more flexibility and variability in their routes and cargoes. This variability can make the trajectories and port-to-port movements of these sub-segments more difficult to predict, possibly explaining the relative under-performance of these sub-segments in comparison to Capesize.

From Table 4.4, we can observe the APDE and MPDE for each ship segment and number of days before arrival. These metrics provide insight into the performance of the model when it comes to predicting the precise location of the ports, quantified by the Haversine distance between the true and predicted ports. We also note that the PFD accuracy is quite similar to the top 1 accuracy for Capesize vessels, but for VLOC, Supramax and Panamax the accuracy improvement is larger. VLOC has an increase of 8.69 percentage points, which is quite significant, indiciating that one should consider the PFD as a decision strategy for these vessels.

As the number of days before arrival increases, we generally see an increase in both the APDE and MPDE. This is expected, as the further in advance the prediction is made, the higher the associated uncertainty tends to be. The most accurate predictions are made closest to the day of arrival at the port, as indicated by the lowest APDE and MPDE values typically being observed one day prior to arrival.

For the Capesize sub-segment, specifically, the APDE and MPDE for 'All days' are 948.5 km and 447.9 km, respectively. As the days before the arrival increases, these errors gradually increase, indicating that the model's ability to predict accurately diminishes. For instance, the APDE and MPDE increase to 1323.8 km and 717.2 km, respectively, seven days before arrival. A histogram of the distances for Capesize 'All days' is displayed in Figure 4.9, where we can clearly see that most wrong port predictions are fairly close to the true port. This indicates that the method often predicts ports in the same vicinity, which will be further emphasised later in the cluster prediction analysis.

The performance disparities between the VLOC, Capesize, Panamax, and Supramax sub-segments could also be explained by the size of their respective response spaces. The VLOC and Capesize sub-segments, with the smallest response spaces, demonstrated the highest prediction accuracies. Conversely, the Supramax sub-segment,

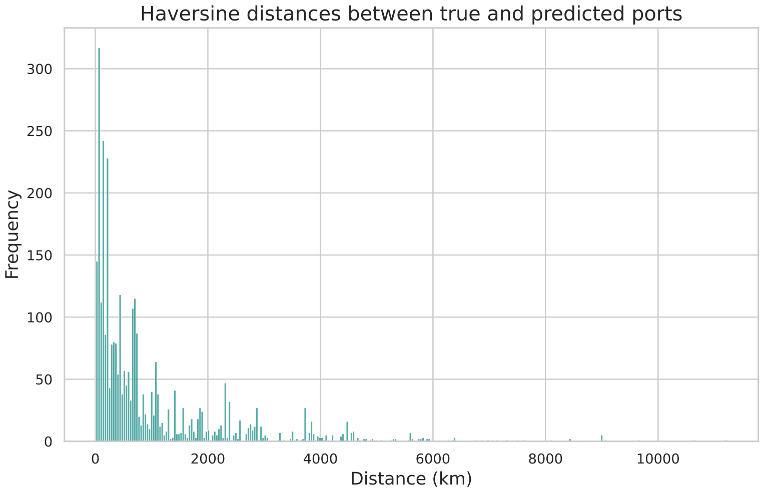


Figure 4.9: Histogram of Haversine distances between true and wrongly predicted ports

with the largest response space, yielded the lowest accuracies. The size of the response space is of critical significance when dealing with prediction models such as XGBoost (Wan et al. 2021). A smaller response space implies that the model has fewer potential outcomes to discriminate between, which can often lead to higher prediction accuracy. In our case, the VLOC sub-segment, having the definite smallest response space with 55 ports, could make it easier for the XGBoost models to correctly identify the destination port. Fewer potential ports can simplify the decision boundary that the model must learn, leading to a more accurate model. On the contrary, the Supramax sub-segment, with the largest response space, presents a more complex decision problem for the XGBoost models. The model must distinguish between more possible ports, which can lead to a more complicated decision boundary and potentially lower accuracy.

This is also clearly seen in the Cluster and Country prediction, which are consistently better for all possible combinations. Regarding the cluster predictions, the results indicate a generally higher accuracy compared to the top 1 port prediction. This is expected given that predicting the cluster of destination ports is a less granular task than predicting the exact port, and thus less prone to errors. As observed in Table 4.2, using only static features, the Capesize sub-segment reached a cluster accuracy of 44.7%, higher than its top 1 accuracy of 36.87%. When both static and trajectory features were used (Table 4.4), the Capesize sub-segment achieved even higher cluster accuracies, reaching up to 77.7%.

As for the country predictions, the results follow a similar pattern to the cluster predictions, with even higher accuracies. This is again consistent with the expectation that a broader category (i.e., country) should be easier to predict than a more specific one (i.e., port or cluster). For instance, when both static and trajectory features were employed, the Capesize sub-segment achieved a maximum country accuracy of 92.3%. The VLOC outperformed this as well, with a country accuracy of 96 % in the same scenario, which is largely attributed to the small response space and predictability between large importers and exporters, shown in Figure 4.10.

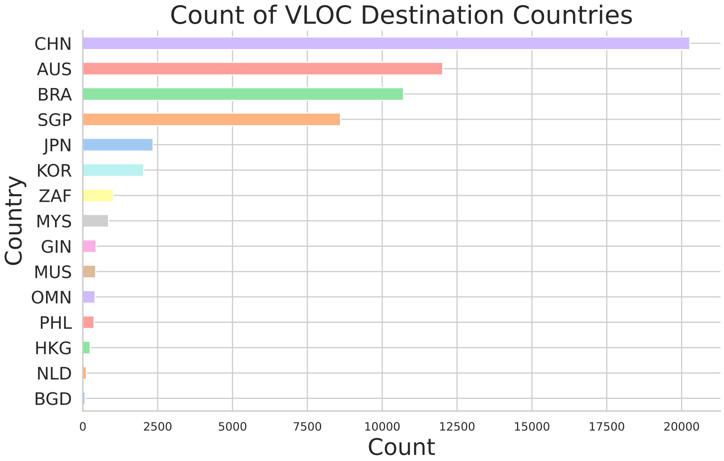


Figure 4.10: The distribution of countries for port-to-port VLOC voyages.

4.2.4 Port-to-port feature importance

Given the multifaceted nature of our investigation, and considering the sheer number of machine learning models developed (36 in total), it was necessary to select a representative model for a more in-depth analysis of feature importance. The Capesize all-features model was chosen for this purpose for a few reasons. First, Capesize vessels are among the largest dry bulk carriers, meaning they are responsible for a significant proportion of global dry bulk trade, making their voyage patterns particularly influential within the shipping industry. Second, the 'all-features' model leverages the full spectrum of available data - both static and trajectory-based features - thereby allowing us to explore a more complete and nuanced understanding of feature importance. This would not be possible with models based solely on static or trajectory-based features. Lastly, the Capesize all-features model displayed a clear tendency towards overfitting (as shown in Figure 4.1), providing an opportunity to investigate whether certain features might be contributing disproportionately to this behavior, and thus might need to be addressed to improve model generalizability.

The boxplots in Figure 4.11 display the PFI for the Capesize model with static and trajectory features, providing a visual representation of the influence of each feature on the prediction of voyage destinations. From these results, it is apparent that certain features have a significantly higher impact on the model's predictions than others. The 'MLTD_lat', 'end_lon', 'MLTD_lon', and 'end_lat' features - which are related to the vessel's trajectory and the most likely trajectory destination (MLTD) - emerge as the most important. The 'end_lat' is the final latitute coordinate sent from the AIS signals. This finding aligns with the nature of shipping routes, which are largely determined by geographical coordinates. The longitude is more important than the latitude, which might be because of the wider span(-180 to 180 versus -90 and 90), and also that there are many latitude values that never appear, for instance values close to the poles.

The model may be over-reliant on the MLTD features, fitting tightly to these predictions in the training data and hence performing less well on unseen data. This observation suggests a potential avenue for improving the model's performance, such as by reducing the reliance on MLTD features or by incorporating regularization tech-

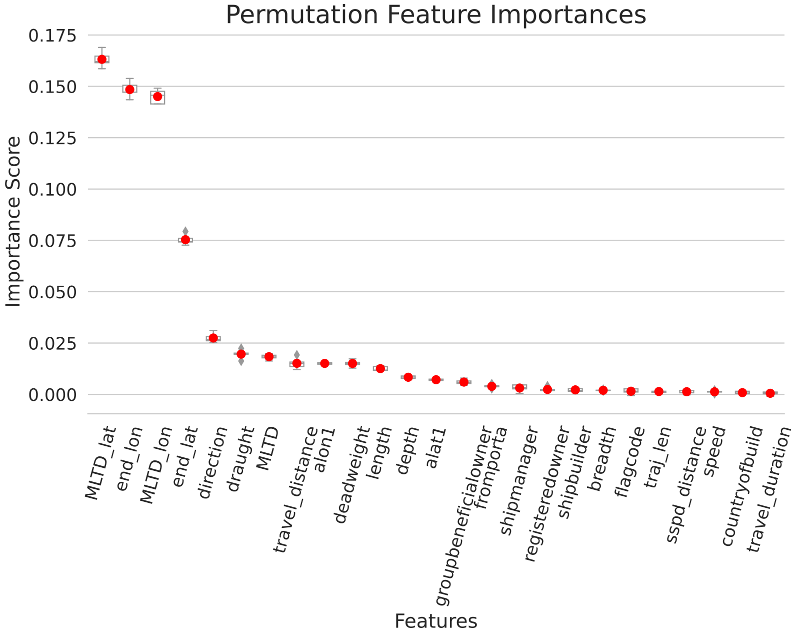


Figure 4.11: PFI for Capesize port-to-port predictions with static and trajectory features

niques to mitigate the overfitting. However, implementing regularization techniques with XGBoost only slightly improved the results, as mentioned above.

The remaining features display a range of importance values, with trajectory-related features (e.g., 'direction', 'travel_distance'), ship characteristics (e.g., 'draught', 'deadweight', 'length', 'depth', 'breadth'), and administrative information (e.g., 'groupbeneficialowner', 'fromporta', 'shipmanager', 'registeredowner', 'shipbuilder', 'flagcode', 'countryofbuild') playing a role in the prediction of voyage destinations. However, their individual impact on the model's output is substantially less than that of the MLTD and end-coordinate features.

4.2.5 Using AIS destination field as a feature

As mentioned in the introduction, the AIS signal contain a manually inputted destination field. This data is frequently marred by inconsistencies and inaccuracies due to human error during entry. It could include spelling mistakes, abbreviations, aliases, or even special characters, which can lead to confusion and misinterpretation of the vessel's actual destination (Abdallah et al. 2019). The destination field in this dataset is thoroughly preprocessed by AF Code. The first step in this process is removing unnecessary noise from the data, such as special characters. Special characters are often used in typing errors or as a workaround for symbols and terms not easily represented with a standard keyboard. However, these characters can muddle the dataset and complicate data analysis.

Next, the destination field data is matched against a reference table. This table contains an exhaustive list of all recognized port names, which have been standardized for consistency. It includes alternative designations, abbreviations, and common misspellings of each port name. By cross-referencing the manually inputted data with this table, AF Code ensures that each destination aligns with an existing port.

For port-to-port Capesize voyages, the destination field is correct 46.73% of the time, which is over 10% worse than for the machine learning results for Capesize, and roughly the same as for the MLTD for Capesize. However, by using this destination field as a feature, one could harness the predictive power of both the MLTD and the destination field, along with the other features.

The results for the same model, but with the added destination feature, are displayed in Table 4.5. The results are significantly better than without using the destination as a feature, with an accuracy of 74%, indicating that adding the destination field as a feature significantly boosts the predictive power of the model. For the laden and ballast voyages, the destination field is therefore part of the initial analysis. The PFI for this model is shown in Figure 4.12, where one can see that the destination feature is clearly the most influential feature.

Table 4.5: Prediction Results for Port-to-Port with the destination feature, along with static and trajectory features

Segment	**Days before**	**Accuracy** (%)					**Error**		
		Top 1	**Top 3**	**PFD**	**Cluster**	**Country**	**APDE(km)**	**MPDE(km)**	**Brier Score**
Capesize	All days	73.99	89.35	74.48	80.51	92.78	1048.94	371.19	0.364
	1	79.24	91.56	-	-	-	983.55	388.36	0.293
	2	76.08	89.82	-	-	-	892.68	371.19	0.330
	3	71.98	88.83	-	-	-	851.65	325.80	0.398
	7	62.57	84.73	-	-	-	1345.83	471.24	0.521

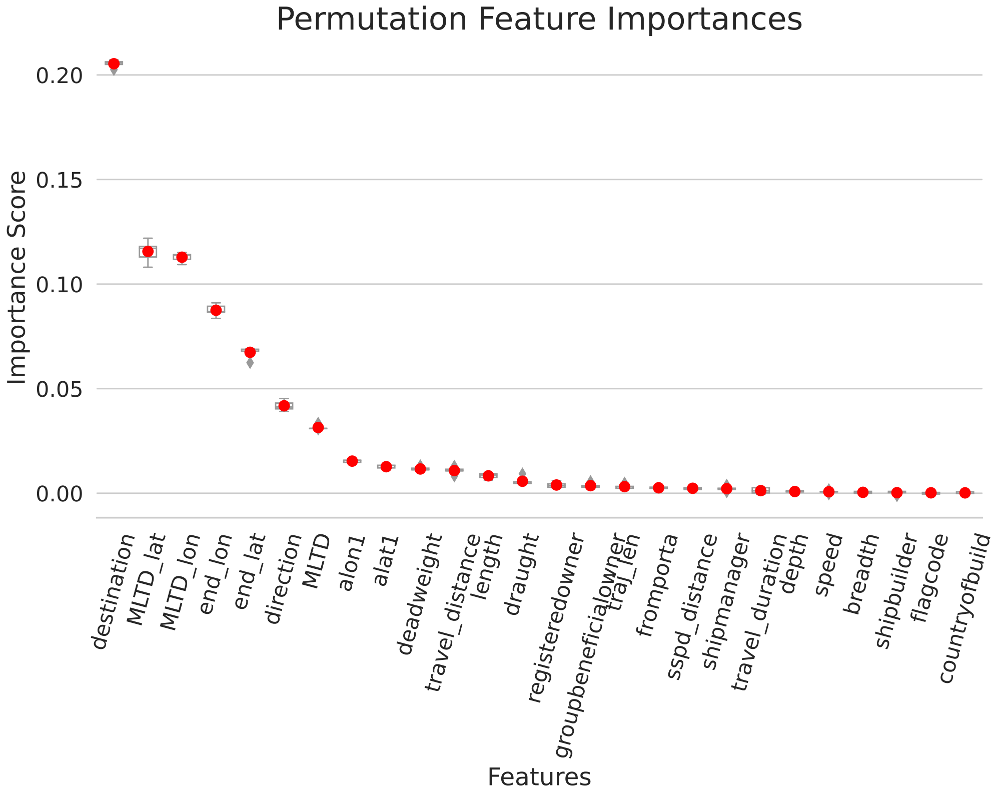


Figure 4.12: PFI for Capesize port-to-port predictions with static, trajectory and destination features

4.3 Laden voyages

The methodology of temporal segmentation for the laden voyages differs, as it focuses on the time elapsed after departure rather than the time remaining before arrival. This change in perspective necessitates a different presentation of the temporal results compared to the port-to-port results, displaying accuracy as a global measure but also as a function of time passed since departure. It is important to note that due to this alteration in the nature of the prediction problem, and also due to that the voyages themselves are different, the outcomes should not be directly compared with those of port-to-port predictions.

4.3.1 MLTD for laden voyages

The MLTD and MTLDC accuracy for laden voyages, along with weighted harmonic mean of precision and recall, are found in Table 4.6.

Table 4.6: Accuracy and F1 scores

Metric	MLTD	MLTDC
Accuracy	0.2095	0.7519
F1 Weighted	0.1915	0.7233

When considering the F1 weighted scores, which take into account precision and recall, the MLTD model achieves a score of 0.1915. Conversely, the MLTDC model obtains a higher F1 weighted score of 0.7233.

The notable difference in accuracy and F1 weighted scores between MLTD and MLTDC can be attributed to the fact that the MLTD model primarily struggles with port prediction, while the MLTDC demonstrates better performance in determining the correct country associated with a given port. This indicates that the method often predicts the correct country but faces challenges in accurately predicting the corresponding port. This can be further explored by looking at the confusion matrix for the top 20 ports in this dataset, displayed in Figure 4.13. One can see that for most of the ports in China, the recall is quite low, indicating that the MLTD method for laden voyages struggles to predict these ports correctly, since it often predicts other ports in China instead.

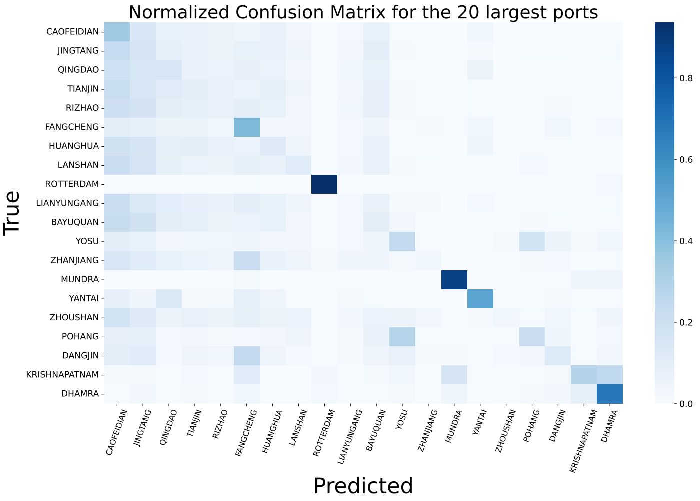


Figure 4.13: Confusion matrix of laden voyages normalized to present recall

In Figure 4.14, the wrongly predicted voyages are plotted as red, and the correctly predicted are plotted as cyan. It becomes quite clear from this Figure that the MLTD often incorrectly classifies a lot of ports in China, if one inspects the red lines at around 130 degrees (longitude). The best area seems to be voyages traveling from the west coast of South America, which might be due to specified trade routes that are commonly used.

The proportion of MLTD accuracy grouped by days since departure from the load port is displayed in Figure 4.15. The trend is that the MLTD accuracy increases with time up to a point, which makes sense considering it increases the likelihood that the vessel is getting closer to its final destination. For voyages that last longer than 4 weeks, the

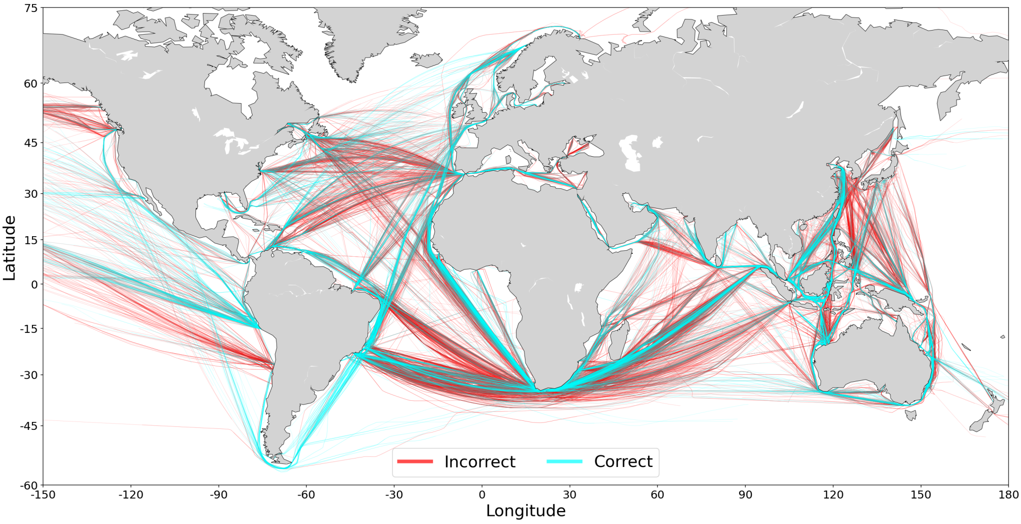


Figure 4.14: Laden voyages on a world map, color-coded based upon correct and incorrect MLTD accuracy

accuracy decreases slightly again, but it is noted that the sample size is much smaller here.

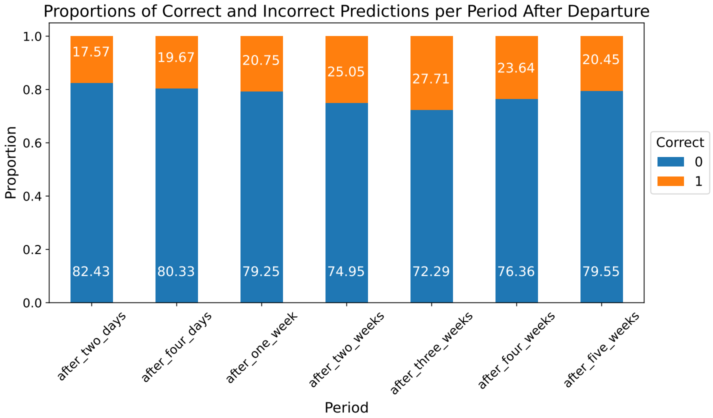


Figure 4.15: Proportion of MLTD accuracy grouped by days since departure

4.3.2 Machine learning for laden voyages

Table 4.7 showcases the prediction results for laden Capesize voyages, using different combinations of feature sets. These feature sets include a combination of Static and Trajectory features with destination, Static features with Trajectory, Trajectory alone, and Static alone. Note that the destination field itself has an accuracy of 44.94% for this dataset.

Table 4.7: Prediction Results for Laden Capesize voyages

Feature set	Accuracy (%)					Error		
	Top 1	Top 3	PFD	Cluster	Country	APDE(km)	MPDE(km)	Brier Score
Static + trajectory + destination	64.65	80.74	64.67	76.95	96.14	1785.56	430.03	0.48
Static + trajectory	44.16	69.06	44.20	58.66	87.50	2376.09	659.56	0.68
Trajectory	24.27	46.28	25.54	41.12	77.64	2540.41	1021.11	0.86
Static	38.02	38.00	38.10	48.97	76.22	2909.70	1080.58	0.76

The model using Static and Trajectory features produced reasonable accuracy rates, with Top 1 and Top 3 accuracies of 44.16% and 69.06%, respectively. However, it resulted in a larger error, with an APDE of 2376.09 km, an MPDE of 659.56 km, and a Brier Score of 0.68.

In comparison, using only Trajectory features led to lower accuracy rates and higher errors. The Top 1 and Top 3 accuracies dropped to 24.27% and 46.28% respectively, and the APDE increased to 2540.41 km, the MPDE to 1021.11 km, while the Brier Score was the highest among all the feature sets at 0.86, indicating poor prediction performance.

Finally, using only Static features had mixed results. Although the top 1 accuracy and PFD accuracy were higher than for only static, the cluster accuracy was the lowest (41.12%). The prediction error distances were also the largest among the feature sets, with an APDE of 2909.70 km and an MPDE of 1080.58 km. The Brier Score of 0.76 is relatively high, indicating poorer prediction quality compared to the other feature sets (except only trajectory features). An interesting observation here is that the static dataset had 12 percentage points higher accuracy than the trajectory set, however the MPDE was higher, indiciating that when it first predicted wrongly, the errors were more severe.

The laden voyages studied encompassed durations from 1 day up to as many as 41 days. The mean accuracy per every other day is displayed in Figure 4.16, color-coded on the amount of voyages that had a record in that day. The accuracy increases with time, which is aligned with the belief that the closer the vessel is to the destination, the easier it is to predict correctly.

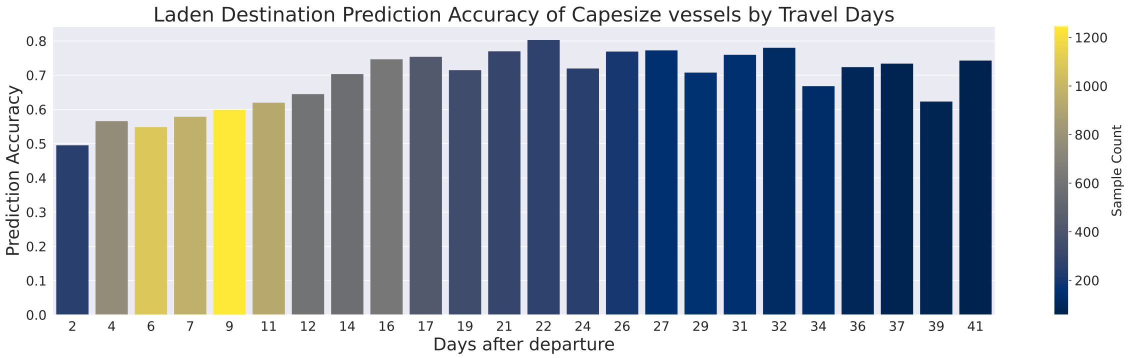


Figure 4.16: Mean accuracy per day traveled for Capesize Laden voyages

4.3.3 Laden voyages feature importance

The PFI for the model using the destination feature show an over-reliance on that particular feature, making it harder to distinguish between the importance of the other features. One possibility would be to log-transform the PFI, however this is might still obscure the results. Therefore the feature importance analysis is done on the model that used static and trajectory features, to ensure comparability. The PFI for laden voyages with static and trajectory features are displayed in Figure 4.17.

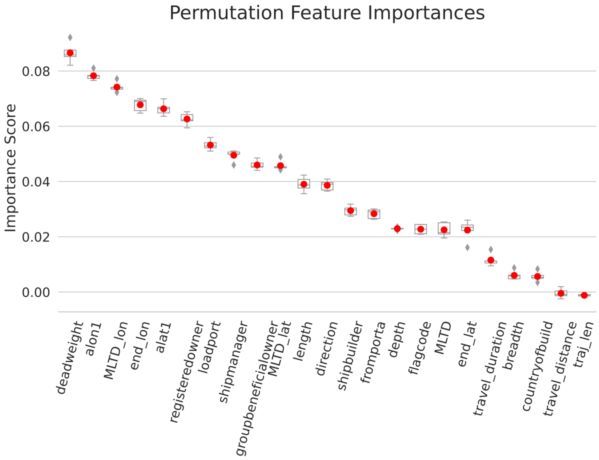


Figure 4.17: PFI for Capesize Laden predictions with static and trajectory features

There seems to be a linear trend for the feature importance with trajectory and static features, contrasting the feature importance for the same feature set for port-to-port voyages in Figure 4.11, where it was overly reliant on the MLTD features. Some static features are now more important, indicating that trajectory-based features might be less important. Considering the nature of laden voyages, where they travel a much greater distance on average than port-to-port (3211 km vs 1878 km), it makes sense that the administrative and static features of the ship instead indicates likely unload destination.

Vessel deadweight and departure port coordinates, along with some MLTD features such as the longitude, emerge as the most important features. The registered owner, group beneficial owner and ship-manager are also three important features, which might correlate to the trade industry as explained in the introduction. As the prediction of laden voyages relies on forecasting the flow of trade goods, it becomes apparent that decisions related to those in charge of trade management carry significant weight in this context.

Due to this, it could be interesting to examine the PFI of the model using only static features. These are displayed in Figure 4.18. The linear trend that came from using trajectory and static features has now been replaced by a more exponential trend, where the coordinates of the departure port now emerge as the most important static features.

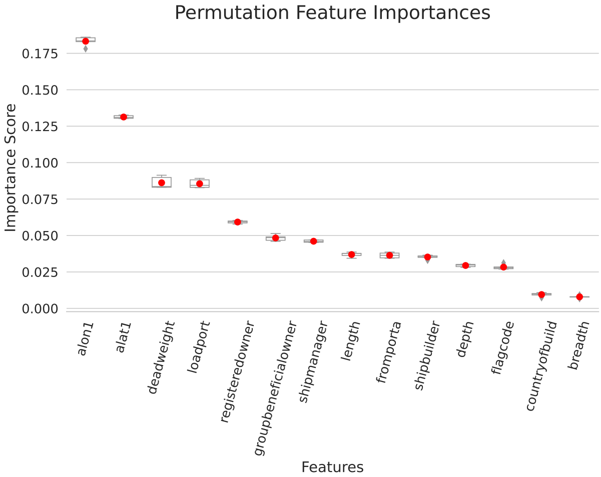
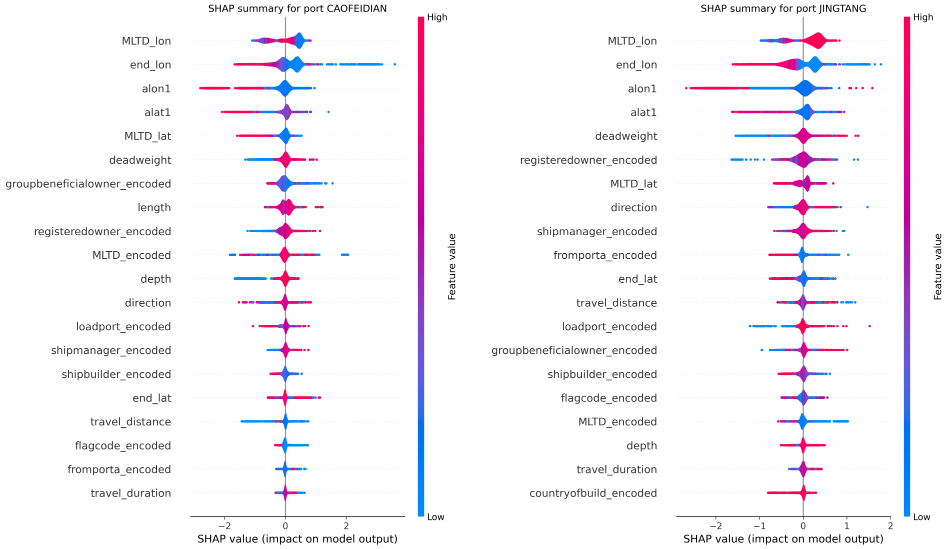


Figure 4.18: PFI for Capesize Laden predictions with only static features

Considering that deadweight was the most important feature for the other model, but only ranks nr 3 here, might indicate a correlation between vessel deadweight and the trajectory-based features. Vessels with a certain deadweight might be more likely to unload goods at certain ports, where the feature importance of deadweight might become enhanced if one knows the last known coordinates, or the MLTD.

To investigate how some ports are affected by the specific features themselves, we compute the SHAP values, and provide SHAP summary plots. An important aspect to note in the SHAP plots are the encoded categorical features. These have been represented through a process of label encoding, which assigns a unique numerical identifier to each category. However, these numerical values are arbitrary and hold no intrinsic order or value, meaning they are merely labels rather than a feature with a natural numerical interpretation. Consequently, while they appear in our SHAP summary plots, their contribution should not be considered in a quantitative analysis as the model's response to these features is not based on their encoded numeric value. In practice, they are more useful for differentiating between distinct categories rather than evaluating their influence on the model's output.

Since destination is such an important feature, the SHAP values are found for the model without it, to see which other features affect the prediction. The most common laden voyage destination ports are Caofeidian and Jingtang, both located on the east coast of China. The SHAP summary plots for both are displayed in Figure 4.19.



(a) Caofeidan

(b) Jingtang

Figure 4.19: SHAP summary plots for the two most common unloading ports

For Caofeidian, one can see that the feature end_lon, which is the final available longitude coordinate in the trajectory, has a positive effect on the prediction if it is low, meaning it is far west. The same holds for the coordinates of the departure port, thus indicating that voyages to Caofeidian that start far west in South or North America and cross the Pacific ocean are easier for the model to predict correctly. All voyages in the test set heading to Caofeidian are displayed in Figure 4.20, where one can clearly

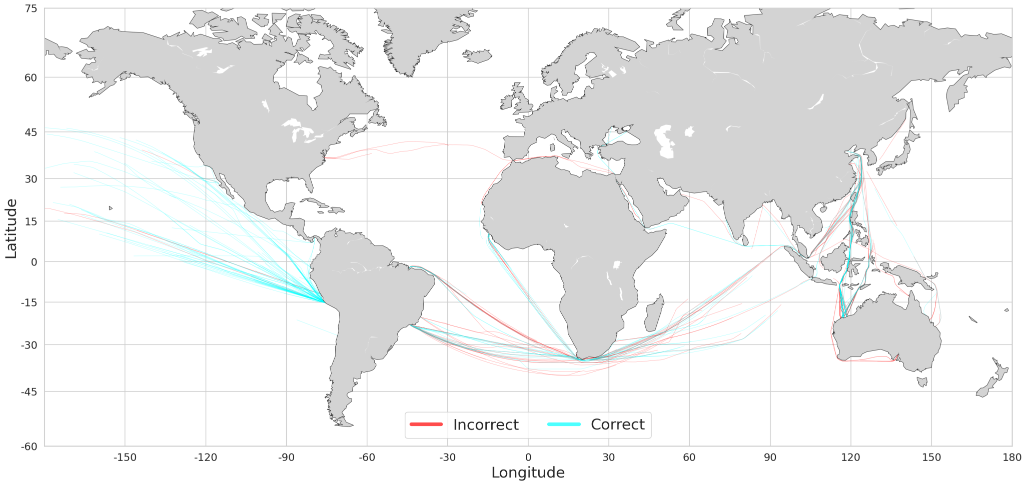


Figure 4.20: Voyages in the test set unloading in Caofeidian, color-coded on prediction. Note that the map has cut off large parts of the pacific ocean, in the western hemisphere.

see this. The model is a lot more confused with voyages traveling from e.g Australia, but from a certain port in Peru, San Nicolas, an iron ore loading port, it almost always predicts correctly. This might be due to certain trade agreements between entities in Peru and Caofeidian. China and Peru entered a Free Trade Agreement in April 2009, the first one between China and a South-American country, which might have influenced this trade route (Angulo-Bustinza et al. 2022).

4.4 Ballast voyages

Ballast voyages, taking place post-unloading and pre-loading, are often shorter than laden voyages, which occur between loading and unloading. The reasons for this are threefold. Firstly, the weight of the ship in the ballast state is significantly less than in the laden state, reducing its draft and allowing for increased speed. Secondly, ballast voyages often follow more direct routing as they are less constrained by port call schedules and fuel considerations. Lastly, operational efficiencies, including efficient bunkering practices, can contribute to the decreased duration of ballast voyages.

4.4.1 MLTD for ballast voyages

The MLTD and MLTDC accuracy for ballast voyages, along with weighted harmonic mean of precision and recall, are found in Table 4.8. The accuracies are significantly higher than for the laden in Table 4.6, as expected.

Table 4.8: Accuracy and F1 scores

Metric	MLTD	MLTDC
Accuracy	0.4712	0.8632
F1 Weighted	0.4250	0.8418

The proportion of MLTD accuracy grouped by days since departure from the load port is displayed in Figure 4.21. There is a clear trend here as well that the MLTD accuracy increases with time, as it did for laden voyages.

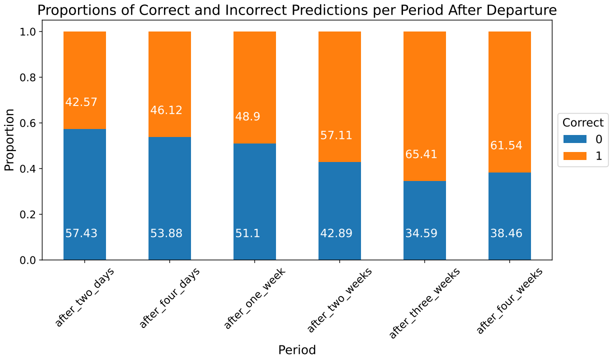


Figure 4.21: Proportion of MLTD accuracy grouped by days since departure for Ballast voyages

A visualisation of this on the world map, zoomed in on the eastern hemisphere, is displayed in Figure 4.22, where the trend clearly is shown with more red (incorrect) trajectories for the lower timeframes and more blue (correct) for the higher timeframes. It is also clearly shown the that there are fewer voyages that have a very long duration.

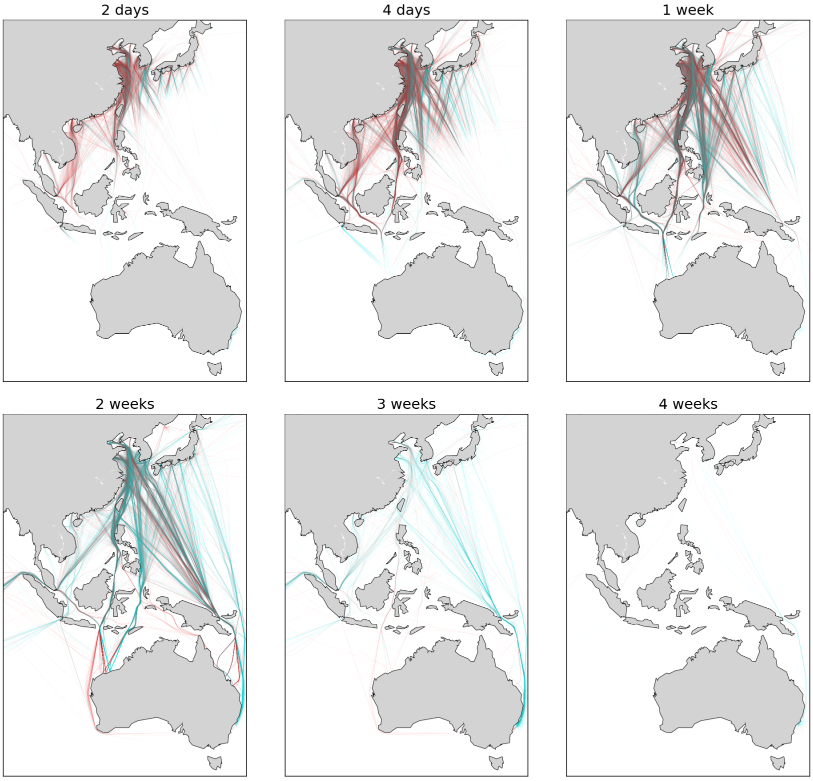


Figure 4.22: Correct and incorrect Ballast voyages for Ballast voyages in the eastern hemisphere.

4.4.2 Machine learning for ballast voyages

The results for the ballast voyages for the different feature configurations are displayed in Table 4.9. Note that the destination field itself has an accuracy of 69.95% for this dataset.

Table 4.9: Prediction Results for Ballast Capesize voyages

Feature set	**Accuracy** (%)					**Error**		
	Top 1	**Top 3**	**PFD**	**Cluster**	**Country**	**APDE(km)**	**MPDE(km)**	**Brier Score**
Static + trajectory + destination	82.58	94.95	82.62	91.36	94.56	2018.13	339.54	0.24
Static + trajectory	66.92	91.78	66.89	83.34	90.69	1725.56	474.68	0.44
Trajectory	53.27	83.34	53.56	77.21	88.26	1610.01	348.81	0.62
Static	65.25	89.83	65.13	77.27	88.13	2648.64	1849.00	0.49

The model with Static + Trajectory + Destination features outperforms the others on almost all metrics, except APDE, where it only beats the model with only static features. The MPDE is lower, meaning that some predictions for the Static + Trajectory + Destination are extremely off, since it increases the average. This could be due to the over-reliance on the destination and MLTD, and if they for some predictions are largely incorrect, it might give some very wrong predictions.

The ballast voyages studied encompassed durations from 3 days up to as many as 41 days. The mean accuracy per every other day is displayed in Figure 4.23.

We observed that the mean prediction accuracy fluctuated across different voyage durations. Voyages after 39 days achieved the highest mean accuracy of over 90%, albeit with a relatively small sample size. The shorter voyage duration of 3 and 4 days had the lowest mean accuracy of around 70%, indicating that it is harder for the model to predict the destination early in the voyages. The general trend suggests an increase in prediction accuracy with the increase in days traveled. This observation aligns logically with the expectation that the model's prediction accuracy improves as the vessel draws closer to its port.

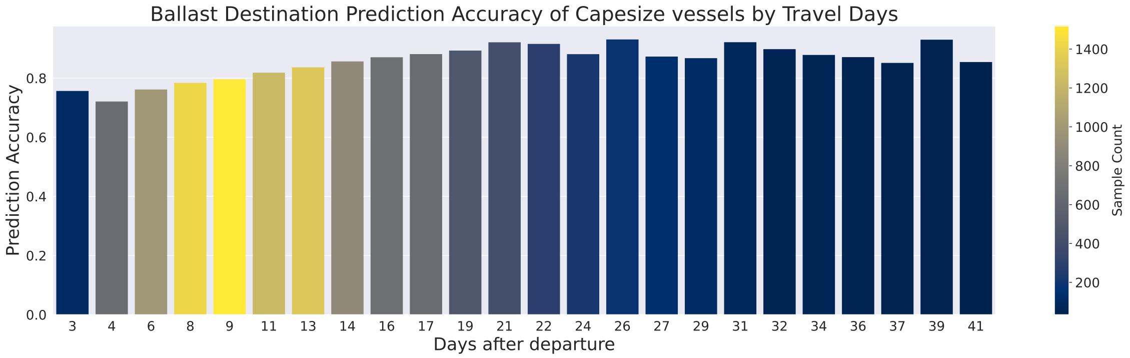


Figure 4.23: Mean accuracy per day traveled for Capesize Ballast voyages

4.4.3 Ballast voyages feature importance

The PFI for the ballast Capesize voyages for static and trajectory features is displayed in Figure 4.24. The end-coordinate features are clearly the most important features according the PFI, especially the final longitude. By randomly shuffling the final longitude coordinate feature, the accuracy drops from 67% down to approximately 40%. In general, compared to the laden voyages, the trajectory-based features seem to be much more important. The vessel deadweight is once again an important feature, ranking as nr 3. This was also seen for laden voyages in the previous section, and might be due to that certain trade routes and port facilities are specifically designed to handle larger vessels, thus limiting the possible ports for vessels with a larger deadweight.

To further investigate how the features themselves affect the predictions, we use SHAP values. We investigate the SHAP values for the three loadports with the highest false positive (FPR) rate. FPR is the proportion of actual negative cases (in this context, incorrectly predicted ports) that are incorrectly identified by the model as positive. Mathematically,

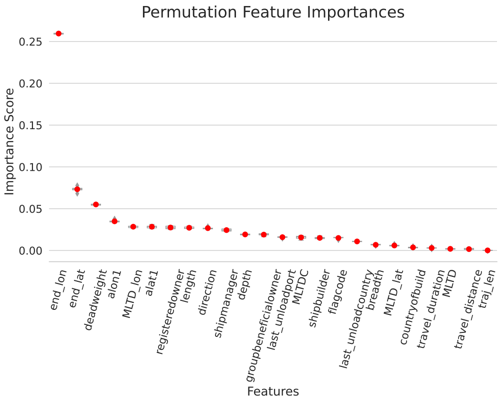


Figure 4.24: PFI for Capesize Ballast predictions with trajectory and static features

$$\text{False Positive Rate (FPR)} = \frac{\text{False Positives (FP)}}{\text{False Positives (FP)} + \text{True Negatives (TN)}}$$

The three ports with the highest FPR, are Port Hedland, Port Walcott and Dampier, with FPR equal 0.92, 0.47 and 0.37. Port Hedland is the most heavily trafficked out of these three ports, which might explain why it has the highest FPR. These three ports are the main source of transporting iron ore from Australia to China (Beresford et al. 2011), and it seems like the model is often confused in this area. The three Australian ports are plotted in Figure 4.25, where one can see the close proximity.

The SHAP values for the three Australian ports are displayed in Figure 4.26. The impact of the ’end_lon’ feature, representing the final longitudinal coordinate of the vessels, is significant. It can be observed from the SHAP plots that a lower value of ’end_lon’ tends to have a lower SHAP value, indicating a lower contribution to the model’s prediction of a positive class. Conversely, a higher ’end_lon’ value often leads to a higher SHAP value, implying a greater contribution to the prediction of a positive class. This suggests that the model has learned a positive correlation between the final longitude of the vessel and the likelihood of predicting a positive class, indicating that vessels that have their final position far east are more likely to travel to one of these

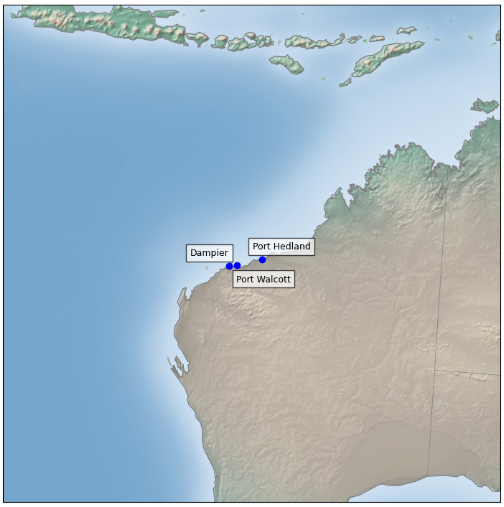
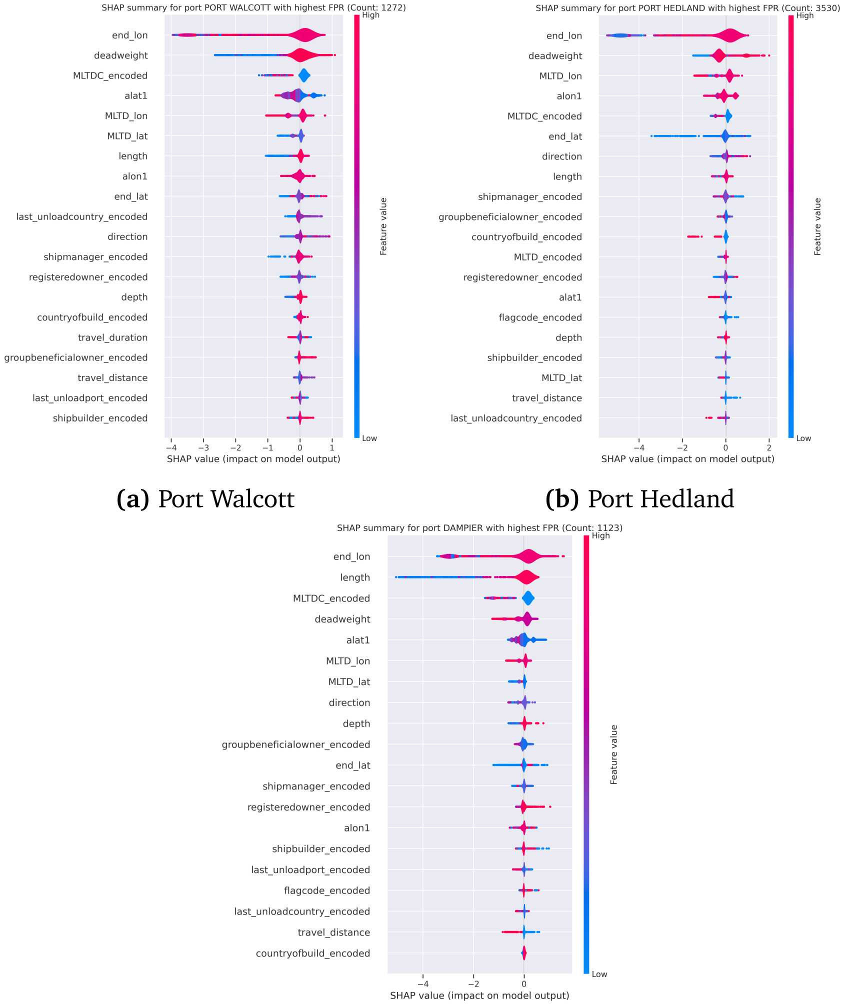


Figure 4.25: Port Hedland, Port Walcott and Dampier

ports.

As mentioned earlier, the iron ore trade between China and Australia is significant, and the three ports analysed here are the main contributors, since e.g Port Hedland is the largest iron ore loading port in the world (Beresford et al. 2011). For all three ports, the features relating to the size of the vessels are all fairly important, where low values for vessel deadweight, length and depth (maximum draught of the vessel) negatively affect the prediction. This means that smaller vessels do not tend to travel to these ports according to the model, which makes sense considering that dry bulk vessels transporting iron ore generally are large (S. Chen et al. 2011). After inspection of port data relating to how large the vessels can be, it was not found a clear correlation between the SHAP values and the draught, max beam and max length of vessels for these ports. Domain experts suggested that the feature importance here is therefore likely influenced by trade factors such as the charterer, which is kept secret.

In general, these type of analyses can then be done for a selected port, to assess the prediction of few voyages. One could inspect the variables that are used for the predictions, compare them with the SHAP values for the predicted port, and see if they align



(a) Port Walcott

(b) Port Hedland

(c) Dampier

Figure 4.26: SHAP summary plots for ports with highest False Positive Rate (FPR)

with the expected values. The overall prediction thereby serves as a comprehensive audit of the prediction model for a new voyage to a selected port.

Chapter 5

Conclusion

This thesis aimed to apply machine learning models for predicting dry bulk vessel destinations in three distinct contexts: port-to-port, laden, and ballast voyages. A significant component of this process involved the collection and compilation of trajectories from AIS data, which necessitated considerable data cleaning due to its extensive and complicated nature.

An initial prediction for the most likely trajectory destination was computed using the Symmetrized Segment Path Distance, which was compared to the Hausdorff distance. It was shown that Symmetrized Segment Path Distance performed better on average than the Hausdorff distance. The most likely trajectory destination prediction performed better in general when the vessel had embarked far on the voyage. The best most likely trajectory destination accuracy was found for Very-Large Ore Carrier vessels, at around 58 %.

The most likely trajectory destination prediction was used as a feature alongside additional factors pertaining to the vessel and its trajectory to enhance the predictive machine learning models. Various combinations of features (static, trajectory, static + trajectory, and static + trajectory + destination) were considered for the models.

Through extensive evaluation and testing of these models on the aforementioned feature sets, it was established that the combination of static + trajectory + destination data provided the most accurate predictions. Metrics such as accuracy, average prediction distance error, median prediction distance error, and the Brier score were used to evaluate the models in this multi-class classification problem.

The models' accuracy ranged from 50-83% for predicting ports, and from 75-96% for predicting countries. The highest performance was seen with larger vessels, notably Very Large Ore Carriers, when it came to port-to-port predictions. The top score achieved was an 83% accuracy rate for Capesize ballast voyages. Interestingly, the prediction of laden voyage destinations proved more challenging compared to those of ballast and port-to-port journeys.

The utility and significance of various features were further examined using permutation feature importance. Moreover, SHAP values were utilized to investigate how particular features influenced predictions for individual classes (i.e., ports). It was found that the influence of certain features on predictions for specific classes was reasonably expected, given the underlying structure of the trade industry. This observation offered invaluable insights into the dynamics of feature importances across different classes.

For laden voyages, the features of the two most common unloadports, Caofeidian and Jingtang, were investigated with SHAP values, and it was found that the most likely trajectory destination longitude of the vessel was the most important feature. It was also found that nearly all voyages from Peru to Caofeidian were correctly predicted. A free trade agreement, the first between China and a South American country, was entered in 2009, which might explain why this prediction was often correct.

For ballast voyages, the model demonstrated the highest false positive rates for the Australian ports of Port Walcott, Port Hedland, and Dampier. Given that these ports are instrumental in the global iron ore trade, a detailed examination of the predictive model's behavior was conducted here as well. This analysis found that lower values for vessel deadweight, length, and depth tended to reduce the likelihood of a vessel's destination being predicted as one of these ports, implying that the model has learned that these major iron ore loading ports are predominantly visited by larger vessels. The model also revealed a positive correlation between the final longitudinal coordinate of a vessel and its likelihood of predicting a positive class, demonstrating how geographic coordinates can provide important insights into vessel movements.

Looking ahead, there is potential to enhance the performance of the predictive models. For instance, incorporating economic data and seasonal trends could be explored as avenues for improvement. Furthermore, refinement of the most likely trajectory destination calculations could potentially yield more accurate predictions. One approach not covered is using a machine learning model to first predict the amount of days remaining of a voyage, and then use a recurrent neural network to predict the upcoming trajectory points, and then finally map the last one of these to the closest port.

As maritime operations continue to evolve and more data becomes available, the predictive models will likely improve, aiding decision-making in this important sector.

Bibliography

Abdallah, N. B., Iphar, C., Arcieri, G., & Jousselme, A.-L. (2019). Fixing errors in the ais destination field. *Oceans 2019-Marseille*, 1–5.

Agnolucci, P., Smith, T., & Rehmatulla, N. (2014). Energy efficiency and time charter rates: Energy efficiency savings recovered by ship owners in the panamax market. *Transportation Research Part A: Policy and Practice*, *66*, 173–184.

Alizadeh, A. H., & Nomikos, N. K. (2013). An overview of the dry bulk shipping industry. *The Handbook of Maritime Economics and Business*, 349–384.

Altmann, A., Toloşi, L., Sander, O., & Lengauer, T. (2010). Permutation importance: A corrected feature importance measure. *Bioinformatics*, *26*(10), 1340–1347.

Angulo-Bustinza, H., Arce-Larrea, G., Calderon-Contreras, V., & Florez-Garcia, W. (2022). Peru-china international trade and its effect on inclusive economic growth in peru 2000-2019. *Decision Science Letters*, *11*(4), 379–390.

Beresford, A., Pettit, S., & Liu, Y. (2011). Multimodal supply chains: Iron ore from australia to china. *Supply Chain Management: An International Journal*, *16*(1), 32–42.

Bergstra, J., Bardenet, R., Bengio, Y., & Kégl, B. (2011). Algorithms for hyper-parameter optimization. *Advances in neural information processing systems*, *24*.

Besse, P. C., Guillouet, B., Loubes, J.-M., & Royer, F. (2016). Review and perspective for distance-based clustering of vehicle trajectories. *IEEE Transactions on Intelligent Transportation Systems*, *17*(11), 3306–3317.

Brier, G. W., et al. (1950). Verification of forecasts expressed in terms of probability. *Monthly weather review*, *78*(1), 1–3.

Carlson, L. (1953). Luleå and narvik: Swedish ore ports. *Journal of Geography*, *52*(1), 1–13.

Chen, S., Frouws, K., & Van De Voorde, E. (2011). Simulation-based optimization of ship design for dry bulk vessels. *Maritime Economics & Logistics*, *13*, 190–212.

Chen, T., & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 785–794.

Chua, J. Y., Foo, R., Tan, K. H., & Yuen, K. F. (2022). Maritime resilience during the covid-19 pandemic: Impacts and solutions. *Continuity & Resilience Review*.

Halden, T. (2019). *Estimation of trade flows with the use of ais data: The case of lng shipping* (Master's thesis). NTNU.

Jin, H. J., Koo, W. W., Sul, B., et al. (2006). The effects of the free trade agreement among china, japan and south korea. *Journal of Economic Development*, *31*(2), 55.

Jing, L., Marlow, P. B., & Hui, W. (2008). An analysis of freight rate volatility in dry bulk shipping markets. *Maritime Policy & Management*, *35*(3), 237–251.

Joseph, A., & Dalaklis, D. (2021). The international convention for the safety of life at sea: Highlighting interrelations of measures towards effective risk mitigation. *Journal of International Maritime Safety, Environmental Affairs, and Shipping*, *5*(1), 1–11.

Jugović, A., Komadina, N., & Perić Hadžić, A. (2015). Factors influencing the formation of freight rates on maritime shipping markets. *Pomorstvo*, *29*(1), 23–29.

Jullum, M., Løland, A., Huseby, R. B., Ånonsen, G., & Lorentzen, J. (2020). Detecting money laundering transactions with machine learning. *Journal of Money Laundering Control*, *23*(1), 173–186.

Kendall, L. C. (2012). *The business of shipping*. Springer Science & Business Media.

Laier, P. (2019). *Vale stock plunges after brazil disaster; $19 billion in market value lost*. https://www.reuters.com/article/us-vale-sa-disaster-stocks/vale-post-disaster-stock-plunge-erases-14-billion-in-market-cap-idUSKCN1PM1JP

Lee, J. M.-y., & Wong, E. Y.-c. (2021). Suez canal blockage: An analysis of legal impact, risks and liabilities to the global supply chain. *MATEC web of conferences*, *339*, 01019.

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in neural information processing systems*, *30*.

Michail, N. A., & Melas, K. D. (2020). Shipping markets in turmoil: An analysis of the covid-19 outbreak and its implications. *Transportation Research Interdisciplinary Perspectives*, *7*, 100178.

Ngoc, N. M., Viet, D. T., Tien, N. H., Hiep, P. M., Anh, N. T., Anh, L. D. H., Truong, N. T., Anh, N. S. T., Trung, L. Q., Dung, V. T. P., et al. (2022). Russia-ukraine war and risks to global supply chains. *International Journal of Mechanical Engineering*, *7*(6), 633–640.

Omholt-Jensen, M. (2021). *Vessel destination forecasting based on historical ais data* (Master's thesis). NTNU.

Papadionysiou, S. (2014). *Analysis of the economics of valemax vessels* (Master's thesis) [Norwegian Open Research Archives]. Norwegian school of economics.

Poļevskis, J., Krastiņš, M., Korāts, G., Skorodumovs, A., & Trokšs, J. (2012). Methods for processing and interpretation of ais signals corrupted by noise and packet collisions. *Latvian Journal of Physics & Technical Sciences*, *49*(3).

Quinlan, J. R. (1996). Learning decision tree classifiers. *ACM Computing Surveys (CSUR)*, *28*(1), 71–72.

Rifkin, R., & Klautau, A. (2004). In defense of one-vs-all classification. *The Journal of Machine Learning Research*, *5*, 101–141.

Roşca, V., Onica, E., Diac, P., & Amariei, C. (2018). Predicting destinations by nearest neighbor search on training vessel routes. *Proceedings of the 12th ACM International Conference on Distributed and Event-Based Systems*, 224–225.

Schubert, E., Sander, J., Ester, M., Kriegel, H. P., & Xu, X. (2017). Dbscan revisited, revisited: Why and how you should (still) use dbscan. *ACM Transactions on Database Systems (TODS)*, *42*(3), 1–21.

Shapley, L. S. (1997). A value for n-person games. *Classics in game theory*, *69*.

Sirimanne, S. N., Hoffman, J., Juan, W., Asariotis, R., Assaf, M., Ayala, G., Benamara, H., Chantrel, D., Hoffmann, J., Premti, A., et al. (2019). Review of maritime transport 2019. *United Nations conference on trade and development, Geneva, Switzerland*.

Stone, M. (1978). Cross-validation: A review. *Statistics: A Journal of Theoretical and Applied Statistics*, *9*(1), 127–139.

Stopford, M. (2008). *Maritime economics 3e*. Routledge.

Taha, A. A., & Hanbury, A. (2015). An efficient algorithm for calculating the exact hausdorff distance. *IEEE transactions on pattern analysis and machine intelligence*, *37*(11), 2153–2163.

Wan, Z., Xu, Y., & Šavija, B. (2021). On the use of machine learning models for prediction of compressive strength of concrete: Influence of dimensionality reduction on the model performance. *Materials*, *14*(4), 713.

Xu, F., Uszkoreit, H., Du, Y., Fan, W., Zhao, D., & Zhu, J. (2019). Explainable ai: A brief survey on history, research areas, approaches and challenges. *Natural Language Processing and Chinese Computing: 8th CCF International Conference, NLPCC 2019, Dunhuang, China, October 9–14, 2019, Proceedings, Part II 8*, 563–574.

Yin, Z., Yang, D., & Bai, X. (2022). Vessel destination prediction: A stacking approach. *Transportation Research Part C: Emerging Technologies*, *145*, 103951.

Zhang, C., Bin, J., Wang, W., Peng, X., Wang, R., Halldearn, R., & Liu, Z. (2020). Ais data driven general vessel destination prediction: A random forest based approach. *Transportation Research Part C: Emerging Technologies*, *118*, 102729.

NTNU

Norwegian University of
Science and Technology