Forecasting mooring duration of ships in the ARA-area and the Rhine based on AIS data

Course: ME44312
Group 1 AIS Data

Julia van Berkom: 4797116 Jasper van den Broek: 5262887 Tanja de Bruin: 6062687
Romy Lambregts: 4881036 Fleur van Steekelenburg: 5313066 David Wolfrat: 4912713

April 12th 2024

ShipSpotting. "TRUDIE" in the Port of Rotterdam. https://www.shipspotting.com/photos/3495226

Abstract

This report explores the relationship between ship characteristics and mooring duration of vessels in the Amsterdam-Rotterdam-Antwerp-Rhine region, as indicated by multinomial logistic regression of AIS data. Three single linear regressions were conducted to analyze the impact of ship draught, length, and width on mooring duration, revealing low R-squared values. Subsequently, a multinomial logistic regression was employed to examine the relationship between ship type and mooring duration. Despite limitations in data quality and size, the logistic regression yielded higher accuracy scores. Results suggest a positive correlation between ship draught and mooring duration, and negative correlations between ship length/width and mooring duration. However, limitations including small dataset size, data inaccuracies, and lack of metadata impact the robustness of the findings, indicating the need for larger, more accurate datasets for further research.

Keywords: AIS data, Maritime ports, Ship characteristics, Mooring duration, Multinomial logistic regression

1 Problem description

For maritime ports it is desirable to operate as efficiently as possible [1]. Port congestion is a main barrier for smooth supply chain management [2]. There is greater pressure on ports because of spatial constraints and ship densities compared to the open sea [3]. For this reason, the need for supervision of ship activities is more urgent, especially when it comes to mooring which refers to a moving vessel that stays within a spatial position for a certain period of time [3]. Therefore, for port authorities, it is useful to gain insights into the movement behavior of ships and more specifically the mooring behavior.

Autonomic Identification System (AIS) is a source of big data for maritime parties and can be used to assess the movements of ships in a port [4]. This system was originally developed for radar augmentation and vessel traffic services. However, AIS data can also be utilised to gather information about traffic in a specific area, thus it allows ships to view marine traffic nearby and to be seen by that traffic [4][5][6].

Although originally designed for radar augmentation, there is a big potential for using AIS data in research [4]. Zhou et al. [6] indicate that ship characteristics can be used as explanatory variables to classify ships into their corresponding behavior clusters. That research only incorporated length and beam as ship characteristics and did not focus on other characteristics. AIS data is also being used for monitoring ship operation congestion [7]. However, Chen et al. [7] concluded that only the size of the ship is not sufficient for estimating the operation time of the ship within the port.

Because of these contradictions in literature, the following research question is determined: *What is the relationship between ship type and mooring duration of vessels using the Rhine and the ARA-ports derived from the provided AIS data?* To answer the main question, different ship characteristics in combination with mooring duration need to be investigated, using single linear regressions and machine learning methods, to find the most accurate assignment of classes. The geographical scope of this research is from the ARA-ports (Amsterdam-Rotterdam-Antwerp) to the Ruhr area. These industrial areas are connected by the Rhine, as stated by Pauli [8] the Rhine is one of the most economical rivers, due to its connection between important ports and the Ruhr area.

AIS data has been used for general behaviour pattern recognition [9], and anomaly detection [10]. Both studies generalise among the vessel types in the data and do not consider different vessel types. Both Sheng et al. [11] and Chen et al. [12] make use of AIS data to classify ship types. They start by grouping the AIS data based on three different operation states. Although the names differ between the papers, the states are based on the type of movement. Chen et al. [12] uses the following labels: static, normal navigation and manoeuvring. The AIS-data used for this report already contained the movement state of the ships. For this report, the 'static' or so-called moored movement state data is used.

Multinomial logistic regression (MLR) is in the literature a widely used method for classification. It has applications in gene prediction [13], computer security [14], digital soil mapping [15] and discovering road network patterns [16]. MLR is used in research due to its high accuracy, simple mathematical model and good interpret-ability [11].

Sheng et al. [11] have used MLR to classify vessels. Based on the trajectory obtained via AIS data, they were successful in classifying fishing boats and cargo ships. Because of the success rates in the existing literature, good interpretability and the fact that this method suits the main research question best, this method will be used in this paper.

This paper is organized as follows. Chapter 2 describes the used data and gives an overview of the preprocessing. Chapter 3 gives an introduction to the machine learning method, and the application thereof. The results will be discussed in Chapter 4. Chapter 5 finalizes the paper with a conclusion and discussion.

2 Data set preparation

This section outlines the process of preparing the AIS dataset for analysis in this research. It begins by describing the collected AIS data and subsequently explains the pre-processing steps undertaken to refine the dataset for regression analysis.

2.1 AIS data set

The AIS data used for this research is collected during a period between the first of January 2021 and the end of January 2021. In total 80.014 AIS data points were collected, after cleaning, pre-processing and filtering, 580 data points were suitable for the regression analysis. These selected points were produced by 9 vessels, ranging from 4 to 168 points per vessel. This pre-processing will be explained in more detail in Subsection 2.2. Table 1 gives an overview of the variables in the complete original AIS dataset with an explanation of the meaning of the variables.

The original dataset is AIS data received every six minutes by the Port of Rotterdam, *if* the vessel was sharing data at that time. Each data point includes all of the variables presented in Table 1.

Table 1: Overview of the given AIS data with an explanation of the different variables

Variable name	**Description**
vessel.name	Name of the vessel
vessel.imo	Unique seven-digit vessel number the company issues to each vessel
vessel.type	Type of vessel (dredging/cargo/et cetera)
vessel.subtype	Hazard level of pressure vessels
vessel.callsign	Unique identifiers to ships and boats for radio transmission
device.mmsi	Maritime Mobile Service Identity number
device.dimensions.to_starboard	Distance of AIS device to starboard
device.dimensions.to_stern	Distance of AIS device to stern
device.dimensions.to_bow	Distance of AIS device to bow
device.dimensions.to_port	Distance of AIS device to port
navigation.status	Status of the sailing vessel (moored/under way using engine/et cetera)
navigation.destination.eta	Estimated time of arrival at destination
navigation.destination.name	Name of the destination
navigation.course	Cardinal direction in which the vessel is being steered
navigation.location.long	Longitude of the vessel
navigation.location.lat	Latitude of the vessel
navigation.heading	Compass direction in which the vessels bow is directed
navigation.speed	Speed at which the vessel is sailing
navigation.time	Time at which the AIS data has been sent
navigation.draught	Depth of the vessel depending on a particular load

2.2 Data pre-processing

To use the AIS retrieved data for regressions to answer the research question, the data has to be pre-processed first. Within this research, this is done in three steps: removing errors, removing unrelated observations and bringing structure to the data.

Firstly, all errors were removed. This was necessary, as numerous vessel and destination names included typing errors, which significantly increased the number of unique vessels. Likewise, many errors could be

found in variables such as navigation.status, with varying amounts of whitespaces. Looking at the unique vessel entries, there are multiple vessels with five or fewer data points, of which was concluded that these points are rather mistakes than valid data. Removing these errors and dropping duplicates resulted in fourteen unique vessels instead of 22. The unique vessel types and amount of observations per vessel type are presented in Table 2.

Table 2: Counts of Vessel Types

Vessel Type	**Count**
Cargo	35541
Dredging-underwater-ops	11678
Tanker	8221
Other	9955

The mean dimensions of the different vessel types seen in the data set are presented in Table 3. These values are calculated based on the given dimensions concerning the location of the AIS device on the vessel.

Table 3: Overview of the means of the characteristics of the different ship types

Ship type	**Length**	**Width**	**Draught**
Cargo	169.8	22.6	6.4
Dredging-underwater-ops	64	16	2.5
Tanker	135	12	1.5
Other	89	11	1.8

The first part of the pre-processing resulted in a refined AIS dataset, depicted by the data points plotted in Figure 1. This figure shows that most data points are located on the Rhine, Westerschelde, the Amsterdam-Rhine Canal and the North Sea.

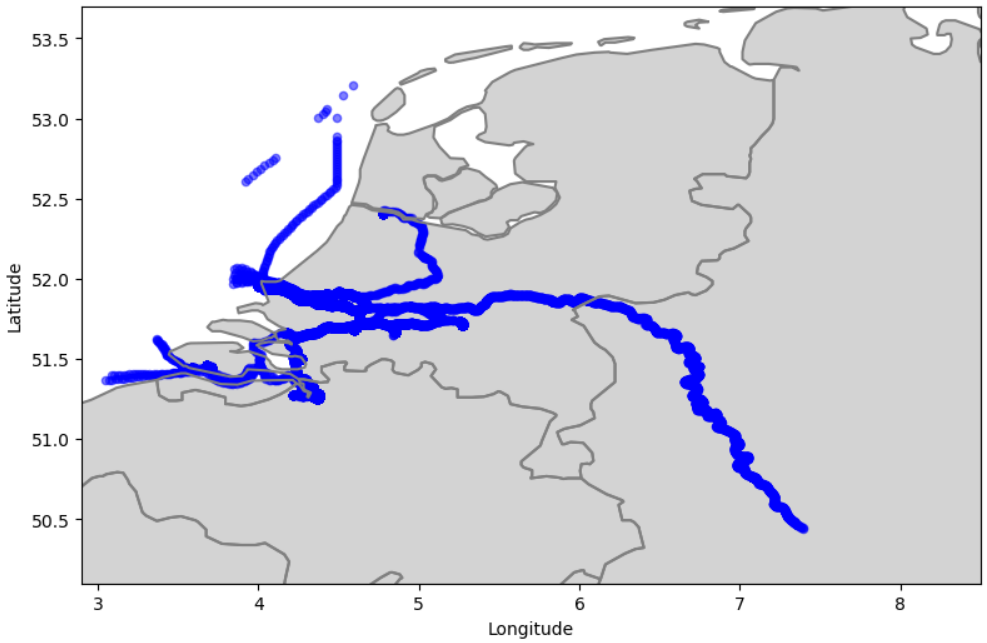

Figure 1: Plot of all the AIS data points

Secondly, the data was filtered to remove unwanted observations within the scope of this research. This mainly targeted the ships that never moor according to their shared data, such as the tanker vessel Charlois.

Thirdly, some data-cleaning was done to bring structure to the data set. All columns were removed except for vessel name, navigation status, vessel type, and vessel draught. Then, the data set was filtered to only show the moored vessels. Three columns were added to the current dataset, the first column contains the time a vessel was moored each time. The last two columns contain the width and length of the ship. The calculated mooring duration, width and length will be used for the regression analysis. This final data set was sorted per vessel, with ascending date of the data points. The final data set included data points on locations presented in Figure 2.

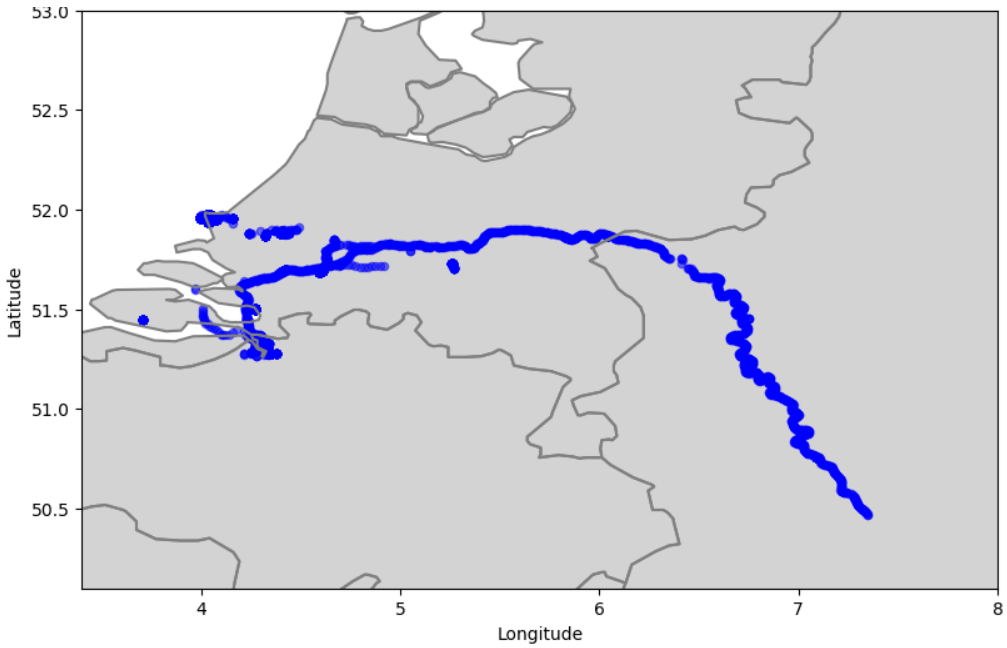

Figure 2: Plot of all the moored data points

3 Methods

In this section, the methods used to find answers to the main research question are elaborated on. First, the linear regression methods are explained, which are used for finding the relationships of ship characteristics on the mooring duration of the ships. Afterwards, a multinomial logistic regression will be described to find the relation between ship type and mooring duration.

3.1 Linear regression

A linear regression is often used to model relationships between variables. Linear regressions are also used to forecast situations [17]. They use independent variables, which are predictors, and dependent variables. In this research, the dependent variable in the linear regression is the mooring duration. Three single linear regressions will be used to find a relationship between the length of the ship and the mooring duration, the width of the ship and the mooring duration and the draught of the ship with the mooring duration. The length and width of the ship are determined in the pre-processing part of the research. Linear regressions are performed using the `LinearRegression` method from the `scikit-learn` package [18] is used in the programming language Python.

The type of the vessel is dependent of more characteristics than only the length, width and the draught of the ship. Therefore, an analysis needs to be done with the variable 'vessel.type', which indicates the type of the vessel. As this variable is measured in categories, a single linear regression does not suit. Therefore, a multinomial logistic regression needs to be conducted as well.

3.2 Multinomial Logistic Regression

As has been demonstrated by Zhou et al. [6], the length and beam dimensions of a ship influence its docking behaviour. However, the different types of cargo that ships may carry require different handling processes. This can result in different mooring durations for the various ship types. Therefore, ship type and mooring duration are investigated as explanatory variables in this study.

A multinomial logistic regression (MLR) is utilised in order to determine the relationship between the type of the ship and the mooring duration of the ship in the given area. An MLR is a classification method where the outcome is multi-class. This means firstly that the dependent variable in this process is nominal, and secondly that there are more than two categories for the data to be ordered into [17]. In this research, the type of the ship will be the dependent variable and is included in three categories: cargo, dredging-underwater-ops and other.

Before the implementation of the multinomial logistic regression, the dataset is split into a train, test and validation set. First, the dataset is split up into train and test data with a ratio of 70% and 30% respectively, based on Vrigazova [19]. Later on, the training data is split up again into training and validation sets with a ratio of 75% and 25% respectively. The training data is assumed more important than the validation data and should therefore be more represented and have a greater share than the validation data. The final ratios for the train, test and validation data are 52.5%, 30% and 17.5% respectively.

For the multinomial logistical regression, the logistic regression classifier package from the `scikit-learn` package [18] is used in the programming language Python. In 3.3 is determined which solver and penalties are used within the Scikit Learn package.

3.3 Algorithm selection and parameter tuning

The `LogisticRegression` class from the `scikit-learn` package provides 9 different combinations of solver algorithm and penalty functions, which supports the multinomial logistic regression [20]. The model fit for each of these combinations is assessed using a maximum of 50 iterations to select the best-performing combination of solver algorithm and penalty function. Next, the best-performing solver algorithm-penalty combination with the best fit is used with more than 50 iterations to investigate whether this improves model fit.

Table 4 shows the results of this analysis. The model with solver 'lbfgs' and without penalty fits best. Therefore this combination is used in the logistic regression function. Increasing the maximum number of iterations to 200 does not further improve the model fit.

Table 4: Model accuracy on the test set of the multinomial logistic regression using the different solver/penalty combinations.

Solver	Penalty	Model accuracy
lbfgs	L2	0.7931
lbfgs	None	0.9943
newton-cg	L2	0.9598
newton-cg	None	0.9885
sag	L2	0.5230
sag	None	0.5230
saga	L1	0.5230
saga	L2	0.5230
saga	None	0.5230

4 Results

In this section, the results of the linear regressions and the multinomial regression will be discussed. First, the results of the three single linear regressions of ship draught, ship length and ship width against the mooring duration are shown. Later on, the results of the multinomial regression with the vessel types are shown.

4.1 Linear regressions

4.1.1 Ship draught

Figure 3 shows a plot of the linear regression where the relationship between ship draught and mooring duration is being modelled. In Figure 3 it can be seen that the spread of the mooring duration (in minutes) is broad for ships with different draughts. Most data points contain ships at draughts below 6 meters. However, the mooring duration for ships at these draughts varies from 0 minutes to about 1400 minutes, which is slightly less than 24 hours of mooring duration. There are only a few data points of ship at deeper draughts (draughts above 6 meters). The ships with a draught of about 16 meters have a mooring duration between 400 and 1400 minutes, which is between 6.5 and 24 hours. These points force the regression line to have the direction as shown in Figure 3. This positive relation can roughly be interpreted as: the deeper the draught of the ship, the longer the duration that a ship has a moored status.

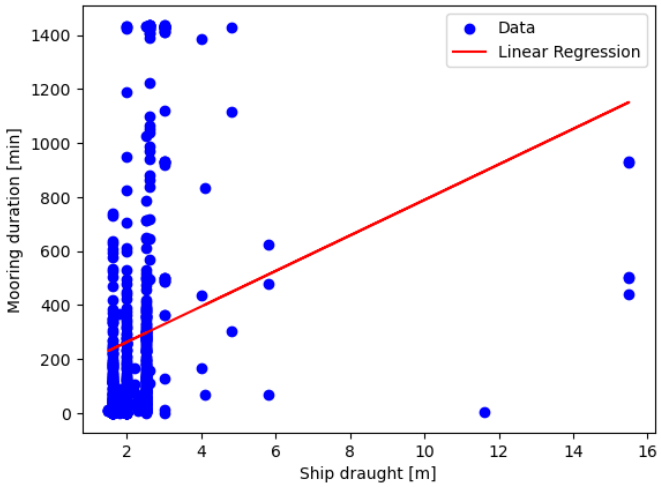

Figure 3: Plot of the linear regression between ship draught and mooring duration

4.1.2 Ship length

Figure 4 shows a plot of the linear regression where the relationship between the length of the ship and the mooring duration is being modelled. In Figure 4 it can be seen that ships with a length between 75 meters and 150 meters have a mooring duration varying from 0 minutes up to 1400 minutes, 24 hours. However, on the length of the ship there are some outliers. There is a ship, the JAN LEEGHWATER, which has a length of 0 meters. Of course, this does not make any sense. There is also a ship with a length of 400 meters which has a mooring duration varying from 400 to 1000 minutes. This ship is much longer than most other ships in the dataset. These two ships force the line of the regression in the current direction, which might differ from the real regression direction. This negative relation can roughly be interpreted as: the longer the ship is, the shorter the mooring duration of the ship is.

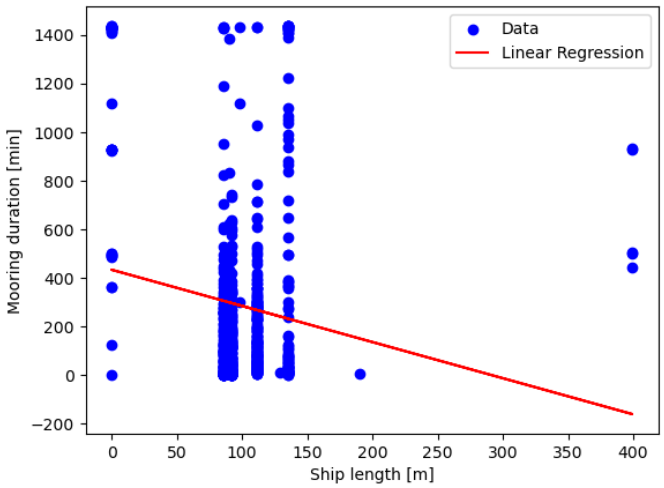

Figure 4: Plot of the linear regression between ship length and mooring duration

4.1.3 Ship width

Figure 5 shows a plot of the linear regression where the relationship between the width of the ship and the mooring duration is being modelled. It shows that most ships with a width between 10 and 15 meters have mooring durations varying from 0 minutes up to about 1400 minutes, 24 hours. This figure also shows that there is a ship included with a width of 0 meters and an outlier with a width of 60 meters. These ships affect the direction of the regression, which could cause a difference from reality. The relationship of the width of the ship and the mooring duration is negative. This can roughly be interpreted as: the wider the ship, the shorter the mooring duration of a ship is.

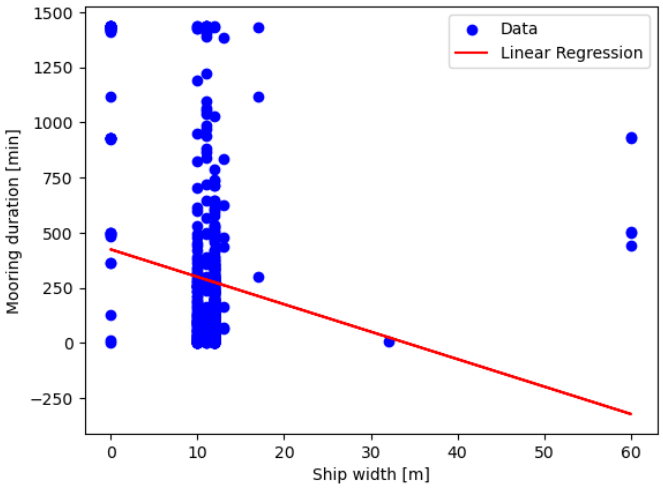

Figure 5: Plot of the linear regression between ship width and mooring duration

4.1.4 Conclusion linear regressions

The linear regressions show that the relationship between the draught of the ship and the mooring duration is positive and that the relationship between the length of the ship and mooring duration and the width of the ship and mooring duration are negative.

Table 5 shows the R-square values of the three linear regressions. The R-square value shows the explained variance of the regression model [17]. The R-square value is usually given as a decimal, but can be interpreted as a percentage. The R-square value for ship draught and mooring duration equals 7.4%. This indicates that the variable ship draught explains 7.4% of the mooring duration. Table 5 also shows that the R-square value of the regression of the ship length and the mooring duration equals 2.4%. This indicates that the variable ship draught explains 2.4% of the mooring duration, which is less than the ship draught. Lastly, Table 5 shows that the R-square value of the ship width and mooring duration equals 3.1%, which indicates that this variable explains 3.1% of the mooring duration of a ship.

Table 5: Overview of the R-squared values of the different linear regressions

Linear regression	**R-square percentage**
Ship draught and mooring duration	7.4
Ship length and mooring duration	2.4
Ship width and mooring duration	3.1

4.2 Multinomial logistic regression

After the single linear regressions, the multinomial logistic regression is conducted. The result of this regression analysis can be seen in Figure 6. This figure shows the three vessel types plotted against the mooring duration in minutes. As can be seen in Figure 6, the ships in the category 'Other' have the smallest mooring duration as the median equals about 100 minutes. However, due to lack of metadata attached to the dataset, it is not known what types of ships belong to this category. This figure also shows that the median of the mooring duration for the category 'Cargo' is a little higher, about 200 minutes. Both categories, 'Cargo' and 'Other' have outliers. The last category, 'Dredging-underwater-ops' has the highest median mooring duration. The mean mooring duration of this category equals approximately 1250 minutes.

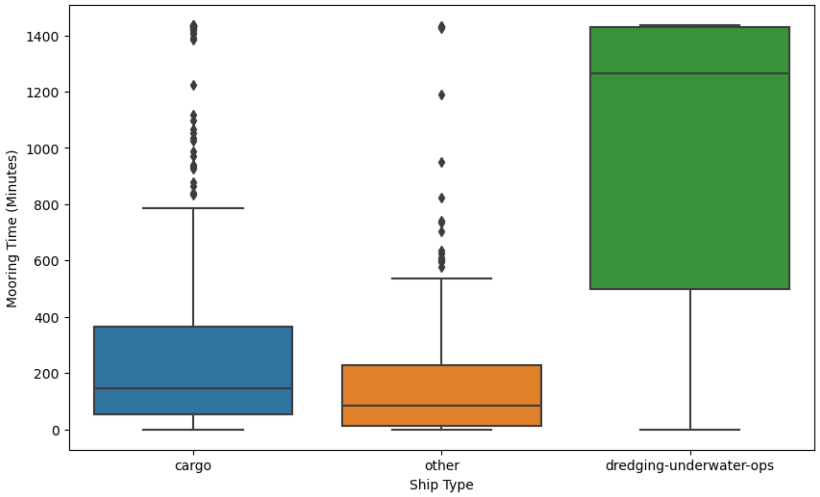

Figure 6: Boxplot of the multinomial logistic regression between the vessel type and mooring duration.

4.3 Classifier

After the model is trained on the training set, the performance of the model is evaluated on the test set, which contains 30% of the dataset. The logistic regression model has an accuracy score of 0.994 on the test set. Figure 8 shows the confusion matrix of the model performance on the test set. One ship of type 'dredging-underwater-ops' is predicted as the 'cargo' type. Furthermore, the prediction matches the true vessel types.

Next to the test set, the model performance is evaluated on the validation set, which contains 17.5% of the dataset. Now an accuracy score of 0.990 is achieved. Figure 7 shows the confusion matrix of the model performance on the validation set. Just like on the test set, one ship of type 'dredging-underwater-ops' is predicted as the 'cargo' type and the rest of the prediction matches the true ship types.

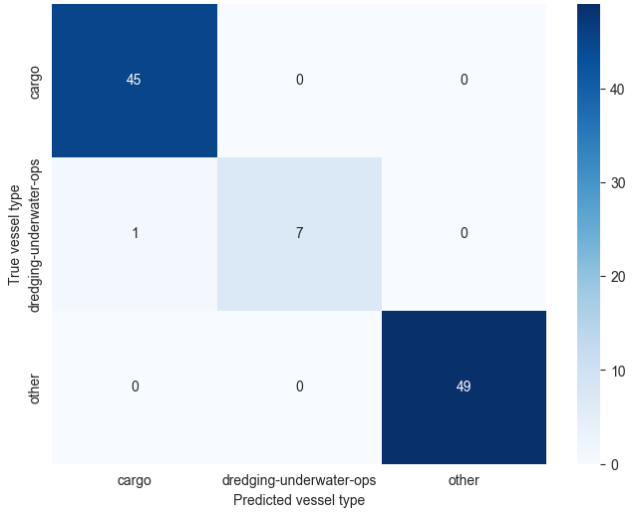

Figure 7: Confusion matrix of the model performance on the validation set.

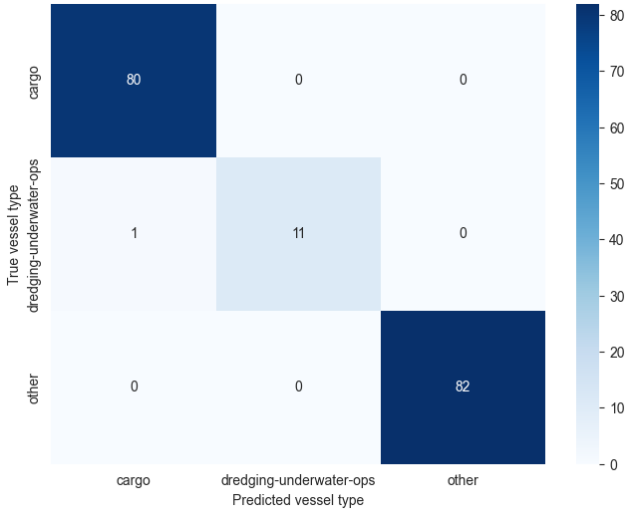

Figure 8: Confusion matrix of the model performance on the test set.

In both situations, one vessel is 'wrong' predicted. However, because the test set contains more data, a higher accuracy score was achieved here.

5 Conclusion and results evaluation

In this section, conclusions will be drawn from the results of the applied methods.

5.1 How to approach produced results

The main question in this research was: *What is the relationship between ship type and mooring duration of vessels using the Rhine and the ARA-ports derived from the provided AIS data?*

For each vessel type it is known what the mean length, mean width and mean draught are in the dataset which was used in this research. In order to find answers to the main question, the single linear regressions are used to explain the relationships between the dimensions of the vessels and the mooring durations. Later on, the multinomial logistic regression is used to predict vessel types based on mooring durations.

5.2 Insights

The results show that there are some relationships between ship characteristics and the mooring duration of a ship. The draught has a positive relation to the mooring duration, which means that an increase in draught results in an increase in mooring duration. The length and width of the ship have negative relations between the characteristics and the mooring duration. This implies that a bigger length or width of the ship, decreases the mooring duration of a ship.

The results also showed that cargo ships have slightly longer mooring durations than ships in the category 'other'. However, the mooring duration of dredging-underwater-ops is far longer than the other two categories. This can be explained by comparing the means of characteristics of the different ship types and the results.

The mooring duration for ships in the category 'Other' is the smallest but not much different from the ships in the category 'Cargo'. Although the cargo ships are much longer and wider than ships in the category 'Other', which implies shorter mooring durations, the draught is also much deeper, which implies longer mooring durations. This balances the slightly longer but less wide ships with a shallower draught of the ships in the category 'Other'. However, the fact that dredging underwater operation ships have the largest mooring duration cannot be explained only by the ship characteristics. The results show that dredging ships are a bit shorter but also a bit wider and they have a slightly deeper draught than ships in the category 'other'.

5.3 Evaluation of the methods

Three single linear regressions were conducted in this research. As previous research [6] has shown relationships between the length and beam of the ship and the classification of behavior clusters, could be found. Therefore, it was assumed that these indicators would suit well to predict mooring behavior. However, the single linear regressions had very small R-square values, which implied that the included variables only explained between 2.4 and 7.4 % of the data. Based on these R-squared values, it is concluded that the dimensions of a vessel can not be used to predict the mooring duration.

Afterwards, the multinomial logistic regression was conducted. This research focuses on predicting the mooring duration based on the vessel type. Unlike the single linear regressions, this model does have a high accuracy score of 99%. This percentage is rather high, but this will be explained in Paragraph 5.5. However, even since this accuracy could be slightly biased, it can be concluded that this model can predict mooring duration based on vessel type.

5.4 Validation

The data is divided into train, test and validation data, 52.5%, 30% and 17.5% respectively. The model is first evaluated and tuned (via hyper-parameter tuning) on the training set. Then the model is evaluated on the smaller validation data set. The accuracy scores on the training and validation set are very close to each other. However, the validation set has an accuracy score of 0.990 which is a little bit lower than the accuracy score of the test set: 0.994. These values indicate that the model fit is excellent.

5.5 Limitations

As can be read in this paper, there are quite some limitations to this research. The first, and probably most important, limitation is the fact that the R-squared and accuracy scores of the linear regression models are very low. The 'best' model has a R-squared value of 7.4% and an accuracy of FIXME, meaning the models do not fit well. Because of this fact, it is hard, if not impossible, to draw the right conclusions from this research. For example, it is not well-founded (enough) to conclude that the width of a ship has a negative impact on the mooring durations of a ship, since the R-squared value is only 3.1% (meaning only 3.1% of the variation is explained). There are different possible reasons regarding the quality of the dataset that may explain these results:

1. The size of the dataset might not be large enough. There are 'only' fourteen unique vessels to start with (after pre-processing) and some of them are also filtered out later on in the process. This is not favorable since it is not very reliable to build a model of 'just' these vessels. A larger dataset would be necessary to build a model with a better fit and accuracy.

2. The provided data in the dataset is not always correct. There are typographical and dimensional errors. For example, the JAN LEEGHWATER vessel does not have the correct length and width input. Also, the TRUDIE vessel states to have a draught of 0 meters, which is very unlikely. These errors, complicate and distort the outcomes of the linear regressions for these dimensions.

3. There is no metadata available for the dataset, thus complicating the implementation of the dataset. For example, the vessel type 'Other' covers quite an amount of data, but due to the lack of metadata, it is not clear what types of vessels are included. Furthermore, the variety of behavior and vessel types within this 'Other' category, could also influence the outcomes. Homogeneous behavior and vessel types would be ideal and make the model more accurate and realistic/trustworthy and less of a problem to group them as one category. However, when those are heterogeneous, it immediately does the opposite to the outcomes. Therefore, metadata would highly improve the dataset.

4. The time range of the data is very short and specific (only January 2021). A different period, e.g. summer or a couple of months, could lead to different input data and thus different outcomes. Also, 2021 was a year in which COVID-19 was highly present. It is not clear whether this affects the data and behavior of the vessels.

5. The locations of the vessels in the dataset are limited. Most vessels sail the Rhine or waters around the Rhine. More various locations, e.g. the IJsselmeer or the North Sea, could lead to a higher variety in vessel types and thus a more extensive dataset.

On the contrary to the 'poor' results of the linear regression models, the results of the multinomial logistic regression are much more accurate, considering the heat map. Even though this is preferable for the results and conclusion, there is still an important remark to make, which again involves the quality of the dataset. Due to the limited amount of vessels, it is not 'surprising' that these vessels are assigned the 'correct' groups. For example, when there are 56 different cargo vessels (instead of the few there are now), chances are higher that some of these will be assigned to the 'wrong' group. This leads to higher

errors and lower R-squared and accuracy values. Concluding that this model is still not very robust, but it is a nice starting point.

Nevertheless the limitations of this research, the models might still give very interesting insights on these topics when larger and more less-error-prone datasets are being used.

References

[1] Mingxiang Feng et al. "Time efficiency assessment of ship movements in maritime ports: A case study of two ports based on AIS data". In: *Journal of Transport Geography* 86 (2020), p. 102741. ISSN: 0966-6923. DOI: https://doi.org/10.1016/j.jtrangeo.2020.102741. URL: https://www.sciencedirect.com/science/article/pii/S0966692318308688.

[2] Xiwen Bai, Haiying Jia, and Mingqi Xu. "Identifying port congestion and evaluating its impact on maritime logistics". In: *Maritime Policy & Management* (2022), pp. 1–18.

[3] Zhaojin Yan et al. "Extracting ship stopping information from AIS data". In: *Ocean Engineering* 250 (2022), p. 111004. ISSN: 0029-8018. DOI: https://doi.org/10.1016/j.oceaneng.2022.111004. URL: https://www.sciencedirect.com/science/article/pii/S0029801822004280.

[4] Martin Svanberg et al. "AIS in maritime research". In: *Marine Policy* 106 (2019), p. 103520. ISSN: 0308-597X. DOI: https://doi.org/10.1016/j.marpol.2019.103520. URL: https://www.sciencedirect.com/science/article/pii/S0308597X18309667.

[5] Wikipedia contributors. *Automatic identification system — Wikipedia, The Free Encyclopedia*. [Online; accessed 12-March-2024]. 2024. URL: https://en.wikipedia.org/w/index.php?title=Automatic_identification_system&oldid=1199241055.

[6] Yang Zhou et al. "Ship classification based on ship behavior clustering from AIS data". In: *Ocean Engineering* 175 (2019), pp. 176–187. ISSN: 0029-8018. DOI: https://doi.org/10.1016/j.oceaneng.2019.02.005. URL: https://www.sciencedirect.com/science/article/pii/S0029801818304761.

[7] Weijie Chen et al. "Monitoring and evaluation of ship operation congestion status at container ports based on AIS data". In: *Ocean & Coastal Management* 245 (2023), p. 106836.

[8] Gernot Pauli. "Sustainable transport: A case study of Rhine navigation". In: *Natural Resources Forum* 34.4 (2010), pp. 236–254. DOI: https://doi.org/10.1111/j.1477-8947.2010.01309.x. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1477-8947.2010.01309.x. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1477-8947.2010.01309.x.

[9] Karl Gunnar Aarsæther and Torgeir Moan. "Estimating Navigation Patterns from AIS". In: *Journal of Navigation* 62.4 (2009), pp. 587–607. DOI: 10.1017/S0373463309990129.

[10] Giuliana Pallotta, Michele Vespe, and Karna Bryan. "Vessel pattern knowledge discovery from AIS DATA: A framework for anomaly detection and route prediction". In: *Entropy* 15.12 (June 2013), pp. 2218–2245. DOI: 10.3390/e15062218.

[11] Kai Sheng et al. "Research on Ship Classification Based on Trajectory Features". In: *Journal of Navigation* 71.1 (2018), pp. 100–116. DOI: 10.1017/S0373463317000546.

[12] Xiang Chen et al. "A ship movement classification based on Automatic Identification System (AIS) data using Convolutional Neural Network". In: *Ocean Engineering* 218 (2020), p. 108182. ISSN: 0029-8018. DOI: https://doi.org/10.1016/j.oceaneng.2020.108182. URL: https://www.sciencedirect.com/science/article/pii/S0029801820311124.

[13] D. McNevin et al. "An assessment of Bayesian and multinomial logistic regression classification systems to analyse admixed individuals". In: *Forensic Science International: Genetics Supplement Series* 4.1 (2013). Progress in Forensic Genetics 15, e63–e64. ISSN: 1875-1768. DOI: https://doi.org/10.1016/j.fsigss.2013.10.032. URL: https://www.sciencedirect.com/science/article/pii/S1875176813000334.

[14] Yun Wang. "A multinomial logistic regression modeling approach for anomaly intrusion detection". In: *Computers Security* 24.8 (2005), pp. 662–674. ISSN: 0167-4048. DOI: https://doi.org/10.1016/j.cose.2005.05.003. URL: https://www.sciencedirect.com/science/article/pii/S0167404805000751.

[15] Wesly Jeune et al. "Multinomial logistic regression and random forest classifiers in digital mapping of soil classes in Western Haiti". In: *Revista Brasileira de Ciência do Solo* 42.0 (July 2018). DOI: 10.1590/18069657rbcs20170133.

[16] Xuesong Wang, Shikai You, and Ling Wang. "Classifying road network patterns using multinomial logit model". In: *Journal of Transport Geography* 58 (2017), pp. 104–112. ISSN: 0966-6923. DOI: https://doi.org/10.1016/j.jtrangeo.2016.11.013. URL: https://www.sciencedirect.com/science/article/pii/S0966692316307001.

[17] Lars Van Heijst. *Regressieanalyse uitvoeren en interpreteren*. nl-NL. Mar. 2023. URL: https://www.scribbr.nl/statistiek/regressieanalyse/.

[18] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[19] Borislava Vrigazova. *The Proportion for Splitting Data into Training and Test Set for the Bootstrap in Classification Problems*. en-US. 2021. URL: https://hrcak.srce.hr/ojs/index.php/bsr/article/view/17347.

[20] Scikit Learn. *1.1. Linear Models*. URL: https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression (visited on 04/11/2024).

A Ship types

Table 6 shows an overview of the characteristics of all the ships in the dataset per ship type.

Table 6: Overview of the characteristics of the different ship types

Ship type	**Length**	**Width**	**Draught**
Cargo	135	11	2.5
	135	11	2.6
	111	12	2.5
	86	11	0.0
	399	60	15.5
	399	60	13.5
	98	17	4.2
	90	13	4.0
	75	8	0.0
Dredging-underwater-ops	0	0	3.0
	64	16	1.9
Tanker	135	12	1.5
Other	92	12	1.6
	86	10	2.0

As shown in Table 6, one ship has a length and width of 0 meters and two ships have a draught of 0 meters. The means are calculated excluding these 0-values and can be seen in Table 3. On average, cargo ships have the highest length, width and draught. Dredging underwater ops have a relatively small length but still a bigger width and a deeper draught than tankers and ships in the category 'Other'.

B Contribution statement

Fleur: I helped with thinking on the method, especially the part of the regression and the plots we needed. Besides, I was responsible for reporting the process. Therefore, I have written parts of the problem description and literature study (Chapter 1). Besides, I have written chapter 2.1 on the explanation of the full dataset, 3.1 and 3.2 on the methods we have used in this research, the whole results part (Chapter 4, except for 4.3) and 5.1, 5.2 and 5.3 on the conclusion parts.

Romy: My job for this project was mainly Python-related. I contributed to building the Python-model regarding pre-processing, filtering, making the plots and help with the multinomial logistic regression. Furthermore, I have also contributed in writing the report. I have written the chapter about limitations and read over/corrected the rest of the report on grammar, content and layout.

David: In this project I mainly worked on the multinomial logistic regression. I worked on how to apply the method and how to program it in Python. I helped with thinking and implementing the preparation of the dataset. My main job was therefore python-related. I was responsible for reporting the multinomial logistic regression part of the report. Therefore I have contributed to the writing of 3.2 (multinomial logistic regression), 3.3 (algorithm selection and parameter tuning), 4.3 (classifier) and 5.4 (validation in the conclusion chapter).

Jasper: Most of my contribution to this project comes from implementing and visualizing the linear regressions and the multinomial logistic regressions, and helping with debugging and cleaning the code.

Tanja: During this project I spend most time on writing the report. Mainly the background literature research included in the introduction and the literature research with regard to the used research methods. Besides I helped with writing or rewriting parts of the report, for example the part about pre-processing. Together with Julia, I spend time on visualizing data and adding data for the final data set. At last, I contributed to the overall checking of spelling and structering the report.

Julia: For this project, I was, together with Romy, responsible for the pre-processing. This included fetching the data, filtering, sorting, making geographical maps of the points and providing the group with the final data set. But also stream-lining between the pre-processing and regression codefiles. In addition, I contributed by reporting this in Chapter 2 and doing a short data analysis. And, like most, I helped with structuring and (grammar-) checking the report.

C Code

This appendix presents most of the code used for this project. Please note that this is not all code, as a lot of the code used is repetition, for example for the plotting of the data. Thus, only the most significant and unique parts of the code are explained here below. The main code was written and run in a Jupyter Notebook file, and data was retracted from different data files in the same GitHub. For reference, all the coding files can be found on GitHub: https://github.com/rlambregts/ME44312-AIS-Assignment.git

```
# Loading and reading data
# Import packages
import os
import json
import pandas as pd
import matplotlib.pyplot  as plt
import folium
from folium.plugins import MarkerCluster
import mplleaflet
import geopandas as gpd
import seaborn as sns

AIS_path = r'.\Data'
AIS_data = []

# Iterate over files in the directory
for file_name in os.listdir(AIS_path):
    # Construct full file path
    file_path = os.path.join(AIS_path, file_name)
    if os.path.isfile(file_path):
        with open(file_path) as f:
            json_data = json.load(f)
            # Append loaded data to the list
            AIS_data.append(json_data)

# Normalize the JSON data
df = pd.json_normalize(AIS_data, 'data')
```

```
# Pre-processing of data
# Drop duplicates based on vessel.imo and navigation.time
df = df.drop_duplicates(subset=['vessel.imo', 'navigation.time'], keep='last')

# Remove spaces
df['navigation.status'] = df['navigation.status'].str.strip()
```

```
# Pre-processing of data
# Delete vessels with 4 or less data points (due to typographical errors)
vessel_name_counts = df['vessel.name'].value_counts()
df = df[df['vessel.name'].isin(vessel_name_counts.index[vessel_name_counts > 5])]

# Function to transform a string into a datetime object
def convert_to_datetime(datetime_str):
    return pd.to_datetime(datetime_str)

# Add new columns with date and time
df['navigation.time'] = df['navigation.time'].apply(convert_to_datetime)
df['date'] = df['navigation.time'].dt.date
df['time'] = df['navigation.time'].dt.time
```

```
# Make new columns based on the change of navigation status and group them
# Get the rows where 'navigation.status' and 'vessel.name' changes
```

```
df['status_change'] = (df['navigation.status'] != df['navigation.status'].shift(1)) (df
                                                ['vessel.name'] != df['vessel.name'].shift(
                                                1))

# Increment group number only when status changes
df['group'] = (df['status_change'] == True).cumsum()
df = df.sort_values(by=['vessel.name', 'group', 'navigation.time'])

# Group by 'vessel.name', 'group', and 'navigation.status' and aggregate start and end
                                                time
result = df.groupby(['vessel.name', 'group', 'navigation.status', 'navigation.draught']
                                                ).agg(start_time=('time', 'first'),
                                                end_time=('time', 'last'), start_date=('
                                                date', 'first'), end_date=('date', 'last'))

# Format the output
for index, row in result.iterrows():
    print(f"{index[0]}: {row['start_date'].isoformat()} {row['start_time'].isoformat()}
                                                    - {row['end_date'].isoformat()} {row['
                                                end_time'].isoformat()} {index[2]}")
```

```
# Calculate the mooring duration
# Calculate time differences between consecutive records
df['time_diff'] = df['navigation.time'].diff()

# Handle cases where navigation.status or vessel.name changes (i.e., start of a new
                                                group)
df['time_diff'] = df['time_diff'].where(df['status_change'] == False, pd.NaT)

# Forward fill NaN values to propagate the time difference across the entire group
df['time_diff'] = df.groupby(['vessel.name', 'group'])['time_diff'].ffill()

df['vessel_length'] = df['device.dimensions.to_stern'] + df['device.dimensions.to_bow']
df['vessel_width'] = df['device.dimensions.to_port'] + df['device.dimensions.
                                                to_starboard']

# Group by 'vessel.name', 'group', and 'navigation.status' and aggregate total time
final = df.groupby(['vessel.name', 'group', 'navigation.status', 'navigation.draught',
                                                'vessel.type', 'vessel_length', '
                                                vessel_width']
                   ).agg(total_time=('time_diff', 'sum'), start_time=('time', 'first'),
                                                            end_time=('time', 'last
                                                            '), start_date=('date',
                                                            'first'), end_date=('
                                                            date', 'last')).
                                                            reset_index()

# Format the output and loop over all lines
for index, row in final.iterrows():
    total_time = row['total_time']
    total_minutes = total_time.total_seconds() / 60  # Convert total time to minutes
    print(f"{row['vessel.name']}: {row['start_date'].isoformat()} {row['start_time'].
                                                isoformat()} - {row['end_date'].
                                                isoformat()} {row['end_time'].isoformat
                                                ()} {row['navigation.status']} - Total
                                                Time: {total_minutes:.2f} minutes")

final['total_time_minutes'] = final['total_time'].dt.total_seconds() / 60
final.drop(columns=['total_time'], inplace=True)

# Only included lines with navigation.status 'moored'.
final_df = final[final['navigation.status'] == 'moored']
```

```
# Perform the linear regressions
from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2_score
import matplotlib.pyplot as plt
import numpy as np
from sklearn import preprocessing

# Make arrays of mooring time and ship depth
mooring_time = np.array(final_df['total_time_minutes'])
ship_depth = np.array(final_df['navigation.draught']).reshape(-1, 1)

# Linear regression
lin_regression = LinearRegression().fit(ship_depth, mooring_time)
print(lin_regression.coef_)
print(lin_regression.intercept_)

y_pred = lin_regression.predict(ship_depth)
r_squared = r2_score(mooring_time, y_pred)
print("R-squared:", r_squared)

# Plot the data points
plt.scatter(ship_depth, mooring_time, color='blue', label='Data')

# Plot the regression line
plt.plot(ship_depth, y_pred, color='red', label='Linear Regression')

# Add labels and legend
plt.xlabel('Ship draught [m]')
plt.ylabel('Mooring duration [min]')
plt.title('Ship draught vs Mooring Duration')
plt.legend()

# Show plot
plt.show()
```

```
# Do the same for ship length
ship_length = np.array(final_df['vessel_length']).reshape(-1, 1)

lin_regression = LinearRegression().fit(ship_length, mooring_time)
print(lin_regression.coef_)
print(lin_regression.intercept_)

y_pred = lin_regression.predict(ship_length)
r_squared = r2_score(mooring_time, y_pred)
print("R-squared:", r_squared)

plt.scatter(ship_length, mooring_time, color='blue', label='Data')

plt.plot(ship_length, y_pred, color='red', label='Linear Regression')

plt.xlabel('Ship length [m]')
plt.ylabel('Mooring duration [min]')
plt.title('Ship length vs Mooring Duration')
plt.legend()

plt.show()
```

```
# Do the same for ship width
ship_width = np.array(final_df['vessel_width']).reshape(-1, 1)

lin_regression = LinearRegression().fit(ship_width, mooring_time)
```

```
print(lin_regression.coef_)
print(lin_regression.intercept_)

y_pred = lin_regression.predict(ship_width)
r_squared = r2_score(mooring_time, y_pred)
print("R-squared:", r_squared)

plt.scatter(ship_width, mooring_time, color='blue', label='Data')

plt.plot(ship_width, y_pred, color='red', label='Linear Regression')

plt.xlabel('Ship width [m]')
plt.ylabel('Mooring duration [min]')
plt.title('Ship width vs Mooring Duration')
plt.legend()

plt.show()
```

```
# Make a boxplot for vessel.type and mooring duration
plt.figure(figsize=(10, 6))
sns.boxplot(x='vessel.type', y='total_time_minutes', data=final_df)
plt.title('Mooring Time per Ship Type')
plt.xlabel('Ship Type')
plt.ylabel('Mooring Time (Minutes)')
plt.show()
```

```
# Perform a Multinomial Linear Regression
# import scikit learn packages for logistic regression
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn import preprocessing
from sklearn.metrics import accuracy_score
from sklearn.metrics import confusion_matrix, classification_report

# Select the right data from the DataFrame
properties = final_df[['navigation.draught', 'vessel_length', 'vessel_width', '
                                        total_time_minutes']]
ship_types = final_df['vessel.type']


# Create training, testing, and validation sets
properties_train, properties_test, ship_types_train, ship_types_test = train_test_split
                                        (properties, ship_types, test_size=0.3)
properties_train, properties_val, ship_types_train, ship_types_val = train_test_split(
                                        properties_train, ship_types_train,
                                        test_size=0.25)

print("Datapoints in Training set:",len(properties_train))
print("Datapoints in validation set:",len(properties_val))
print("Datapoints in Test set:",len(properties_test))
```

```
# Train a logistic regression model using the training set
model1 = LogisticRegression(multi_class='multinomial', solver='lbfgs', penalty=None,
                                        max_iter=200)
model1.fit(properties_train, ship_types_train)

# predict results using the test set
validation = model1.predict(properties_val)

print("For Logistic Regression: ")
print(classification_report(ship_types_val, validation))
```

```
print ("Accuracy of logistic regression on the initial data is: ",accuracy_score(
                                        validation, ship_types_val))

conf_matrix1 = confusion_matrix(ship_types_val, validation)
print(conf_matrix1)

plt.figure(figsize=(8, 6))
sns.heatmap(conf_matrix1, annot=True, fmt='d', cmap='Blues', xticklabels=model1.
                                        classes_, yticklabels=model1.classes_)
plt.xlabel('Predicted vessel type')
plt.ylabel('True vessel type')
plt.title('Confusion Matrix - validation set')
plt.show()

# Calculate model accuracy
prediction = model1.predict(properties_test)
accuracy1 = accuracy_score(ship_types_test, prediction)
print("Accuracy of logistic regression on the extended data is: ", accuracy_score(
                                        ship_types_test, prediction))

# Calculate the confusion matrix
print("Confusion matrix on the test set:")
conf_matrix2 = confusion_matrix(ship_types_test, prediction)
print(conf_matrix2)

plt.figure(figsize=(8, 6))
sns.heatmap(conf_matrix2, annot=True, fmt='d', cmap='Blues', xticklabels=model1.
                                        classes_, yticklabels=model1.classes_)
plt.xlabel('Predicted vessel type')
plt.ylabel('True vessel type')
plt.title('Confusion Matrix - test set')
plt.show()

# Print classification report
class_report1 = classification_report(ship_types_test, prediction)
print("Classification Report:\n", class_report1)
```