

DELFT UNIVERSITY OF TECHNOLOGY

MASTERS'S THESIS

**Unsupervised classification of ships and
their operations**

Author:
Tim CHEUNG

Supervisor:
Prof. dr. ir. Geurt JONGBLOED
dr. ir. Jouke DE BAAR



Unsupervised classification of ships and their operations

by

T. Cheung

to obtain the degree of Master of Science
at the Delft University of Technology

Student number: 4160177

Project duration: February 11, 2019 - January 29, 2020

Thesis committee:

Prof. dr. ir. Geurt Jongbloed TU Delft, supervisor

Dr. Jouke de Baar Damen Shipyards, supervisor

Dr. Dorota Kurowicka TU Delft

Abstract

In this modern age, data is being generated constantly and data is being saved for analysis everywhere. In the maritime industry, interest in the analysis of ship data has grown over the years. In this thesis, we will take a look at AIS data coupled with sea state data.

AIS data is data generated from the ship, concerning the ship locations, speed, and heading, among others. When coupled with data such as the wave height and wave directions at these locations, we can analyse the ship operations in different sea conditions. We analyzed 46 Damen ships of the same type, that operate in different regions of the world. The aim was to make interpretable groups of ships that have similar operation profiles, and to investigate the effect of different sea states on the ship operations.

We first enrich the data with port labels, from which we can define trips as sequences of points away from port. We also estimate path lengths between points using Bézier curves. From this we get a relevant set of variables that can use for an unsupervised learning task.

We clustered the ships using three methods: principal components analysis, K-means, and hierarchical clustering. Principal components analysis showed variation in the ships, but interpretation and definitive clusters were not clear. We then used the K-means method to make 12 clusters of ships, of which six clusters proved to be stable. Hierarchical clustering showed similar results. Interpretation of these clusters was possible, mainly by looking at separate trips. We therefore also clustered the trips, to get classes of trips. We used the K-means method and obtained six clusters of trips, of which five were stable.

We also look at ship availability in different regions during different sea conditions. We use an isotonic regression method to test whether ships stay in port more often during heavy weather. We found regions where availability decreases during high waves and regions where availability seemed independent from wave height. This most likely has to do with the function of the ship. And finally we look at sailing speeds during different sea states and find that sea state data alone is not sufficient to adequately estimate sailing speeds of a ship.

The conclusion is that using the variables that we created, stable clusters can be obtained. These clusters are interpretable and can lead to a better understanding of customer needs. Coupling the AIS data with more data sources however would be a recommendation, since that can lead to more informative clusters, and might lead to more insight into sailing speeds.

Acknowledgements

In this thesis, I will describe the work I have done at Damen Shipyards in Gorinchem. It is the final work of my time as a student in Delft, and a long journey it has been. Starting with a bachelor's degree in Electrical Engineering, venturing out to the faculties of Mechanical Engineering and Technology, Policy and Management, I finally found my bearing again at the faculty of Applied Mathematics. I had a lovely time as a student in Delft, but nevertheless I am glad to conclude the journey with this thesis. There are many people who helped me with this feat, and I would like to take the time to express my gratitude.

In the first place, my sincerest thanks to professor Geurt Jongbloed. From his energetic lectures in statistics that got me interested in the subject, connecting me to Damen, to supervising me the whole way through, it has been a joy to have him as my supervisor. I struggled a lot with finding the right angles to approach this project, but luckily Geurt was always ready to provide me with new ideas to motivate me. Not only did we discuss my work on this thesis, we enjoyed talking about many aspects in life. Looking back, it feels like we talked more about life than anything mathematics related. I learned a lot from both sides of these conversations and for all of this I thank you.

I would also like to thank all my colleagues at Damen. The maritime industry was a mystery to me at beginning of the project, but colleagues at Damen were always available to answer my questions. Special thanks go out to my supervisors at Damen. Thank you to Ewoud Huiskamp for shaping this project, and guiding me through the beginning of the process. Jouke de Baar did a wonderful job picking up the guidance halfway through the project, with much enthusiasm. Even though supervising did not come at the best of times for him, he still always made time for me and did a great job, for which I am grateful. To Don Hoogendoorn, for stepping up to guide my thesis whenever I needed it and providing useful and new insights.

Finally, many thanks go out to my friends. All of them supported me in different ways. My friends from the master's program formed a tight and warm support system, which made studying a lot easier. My other friends helped by listening to my thoughts, discussing the project with me, or simply by providing me with, at times, a much needed distraction from the project. Mostly by climbing with me. For that, I thank all my dear friends.

Tim Cheung
Delft, January 2020

Contents

Abstract	ii
Acknowledgements	iii
1 Introduction	1
1.1 Damen Shipyards	1
1.2 Motivation of the study	1
1.3 Problem description	2
1.4 Thesis outline	2
2 Background	4
2.1 Damen FCS 5009	4
2.2 Data collection	4
2.3 Related research	5
3 Raw data	7
4 Data enrichment	8
4.1 Location-based profiles	8
4.1.1 Areas of interest	8
4.1.2 Status labeling	12
4.1.3 Critiques on labeling	13
4.2 Finding areas of interest based on location density	13
4.2.1 DBSCAN	14
4.3 Speed-based profiles	16
4.3.1 Division into trips	16
4.3.2 Estimating average speeds	19
Interpolation	19
4.3.3 Bézier	21
4.3.4 Low degree Bézier curves	22
Linear Bézier Curves	22
Quadratic Bézier Curves	22
Cubic Bézier Curves	23
4.3.5 Application Bézier	23
4.4 Results Interpolation	25
4.5 Final dataset	34
5 Unsupervised learning	35
5.1 Principal Component Analysis	35
5.2 K-Means	36
5.2.1 Selection of K	37
Elbow method	37
Average silhouette method	39

5.3	Hierarchical clustering	40
5.4	Results clustering	42
6	Sea state influence	43
6.1	Availability of the ship	43
6.1.1	Gulf of Mexico	43
6.1.2	Monotonic regression	47
6.1.3	Persian Gulf	49
6.1.4	The Caribbean	49
6.1.5	Nigeria	49
6.1.6	Conclusion availability of ships	49
6.2	Speed on established routes	49
7	Conclusions	52
8	Recommendations	53
A	All data variables	56
B	Raw data analysis	58
C	Sea state plots without ports	59
D	Bézier curves	60
E	Principal Component Analysis	63
F	Results bootstrap of PCA on the ships	66
G	Isotonic regression	67
	Bibliography	69

Chapter 1

Introduction

1.1 Damen Shipyards

Damen is a family owned shipbuilding company from the Netherlands. It started in 1927 on the River Merwede and today they are still located on the same location, but with customers and service hubs all over the world. Damen builds custom ships, but specializes in delivering ships from standardised designs. This makes for a robust vessel design that is proven to work and by having ships in stock, delivery times are very short.

Damen is active across the entire spectrum of maritime industries. From harbours and terminals to offshore support, from dredging to renewable energy and from superyachts to defence and security, all kinds of ships are being built at Damen Shipyards.

1.2 Motivation of the study

Market segmentation is the activity of dividing a consumer market in subgroups. The consumer market may consist of existing customers, potential customers, or both. The aim is to make subgroups, or segments, where customers share some common characteristics. There is no one right way to make these groups, and the features to base the groups on depend very much on the goal of the segmentation. Goals can for example be marketing, product design, or market exploration.

Emerging technologies these days generate an immense stream of data, and much of it is available for open use. The *Automatic Identification System* (AIS) is one of these technologies. AIS is an automatic tracking system for maritime vessels and, similar to GPS, it provides information about moving vessels such as location, speed, and course. More about AIS can be found in section 2.2.

The business model of DAMEN is to sell ships that are ready from stock, as opposed to taking orders for custom built ships. As a consequence, *Damen* has a high demand for insight into what customers are really looking for. This is not to say that Damen has absolutely no idea. By talking to buyers and by maintaining good customer relationships, wishes can be heard. However, it remains hard to see how a ship is being used after it is delivered to the customer destination. The availability of AIS data however offers the possibility to analyse ship operations as carried out by the customers.

These data can answer some questions that are important to operations within DAMEN. These questions include:

- How do DAMEN ships compare to similar ships from competing companies?

- Is there a difference in sailing speeds?
- Does the availability of ships differ between companies depending on weather conditions?
- Do users of DAMEN vessels change sailing speeds depending on weather conditions?
- Are operations dependent on weather conditions?
- Are the ships being used in the way the designers had in mind?

Answers to these questions can have impact on the way DAMEN currently design their ships. Answers to the first question and subquestions can help the sales department if it can be shown that DAMEN vessels perform better than competitors with a similar target market. Answers to the other three questions are important to the engineers of DAMEN who design the ship. Ideally, sailors are able to navigate the ship at any speed they desire, regardless of the weather conditions. Currently, the ships discussed in this study are being designed in a way such that high waves should have very limited on sailing speeds, but whether customers actually keep constant sailing speeds independent of wave heights is yet unknown.

In this study, we will focus our analysis on a fleet of 46 DAMEN FCS 5009 vessels. This type was chosen because of the comparatively large number of ships sold, and the wide variety of geographical locations the ship is being deployed.

1.3 Problem description

As mentioned in the previous section, we would like to segment the clientele of the DAMEN FCS5009 ships into subgroups with similar characteristics that are of interest. The goal of this research can be stated as follows:

Use statistical techniques on the data to gain useful and interpretable insights in the behaviour of the customers.

This goal poses the challenge of giving meaning to "useful and interpretable". What is useful to one department might be irrelevant to another. This is something we will explore in the coming chapters.

The questions we try to answer in this study are inspired by DAMEN's desire to understand their customers, their behaviours, and their needs. The questions are therefore as follows:

- How can we divide our fleet of DAMEN FCS 5009 ships in groups with similar behaviour within the group?
- Are DAMEN FCS 5009 ship operations influenced by conditions at sea?

1.4 Thesis outline

In the following chapters, we will make an attempt to answer these questions. First, we will have two chapters providing some background into the data, related research, and some exploratory data analysis of the raw data. After that, chapter 4 describes methods with which we enriched our data with extra variables that we believe will help us answer the research questions.

After that, we will address the first question in chapter 5, where we use different methods to make groups of the ships in our data set using the enriched data.

In 6 the second research question is addressed. We look at ship availability and sailing speeds and use statistical methods to determine if these variables are affected by the sea state.

In the final chapters we will conclude our research, and give recommendations for further research.

Chapter 2

Background

In this chapter we will explain the data collection process and review other research projects with similar objectives and datasets.

2.1 Damen FCS 5009

The DAMEN FCS 5009 is a Fast Crew Supplier vessel made by Damen Shipyards, based in Gorinchem in the Netherlands. With a length of 53 meters, a deck area of 240 m² and space for 80 passengers, the vessel is well equipped for fast, safe and comfortable transfer of crew and cargo. Additionally, the vessel is capable of emergency towage and crane handling functionality. The DAMEN FCS 5009 is also sold under the name YS 5009, the Yacht Support. While the two are technically the same, the deck of the yacht support is customized for the purpose of supporting trips of luxurious yachts. It offers room for a helicopter, small speedboats, jet skis and other big machinery that might not fit on the yacht itself.



FIGURE 2.1: The DAMEN FCS 5009

2.2 Data collection

In this project, we will use data gathered from ships fitted with transponders for the Automatic Identification System (AIS). AIS is an automated tracking system that

logs a variety of information about the ship's current state, for example: speed, location (latitude/longitude), course over ground, heading, time, and a unique identification number. Ships with transponders on board can both see AIS data of ships around them, and broadcast their own information simultaneously at regular intervals.

AIS is intended primarily to allow ships to view maritime activity in their vicinity and to be seen by other ships. This provides shippers and port authorities on shore with valuable information for both navigation and collision prevention. It was first widely adopted in 2002 when the International Maritime Organization (IMO), an agency of the United Nations responsible for shipping regulation, agreed on the International Convention for the Safety of Life at Sea (SOLAS) treaty. This agreement included a mandate that required AIS for all ships of 300 gross tonnage and upwards in international voyages, 500 and upwards for cargoes not in international waters and passenger vessels (IMO, 2002). In addition, fishing vessels greater than 15 m sailing in water under the jurisdiction of the European Union Member States shall also be required to be fitted with AIS. In the years following that, AIS products kept evolving and governments instigated projects to endow all types of vessels with AIS technology for improved safety and security. As of 2014, all EU fishing boats over 15m will have to have AIS technology on board (IMO, 2014).

As AIS information is transmitted from the ship, any AIS receiver within range (or satellite) can pick up the signal and save it. MarineTraffic is a company in the marine industry that has many AIS receiver stations and satellites and as such can pick up many AIS signals. They save this data and offer it for purchase for any party that is interested in the data.

Secondly, where AIS data only pertains to the ship itself, we are also interested in the conditions at sea when the ships are sailing. Our vessels are not able to measure weather conditions, so we have to rely on other sources for this data. We use COPENICUS for our sea state data. COPENICUS, previously known as GMES (Global Monitoring for Environment and Security), is the European Programme for the establishment of a European capacity for earth observation and monitoring. Using satellites and marine measuring devices they are capable of measuring various ocean variables. Using these measurements and models, they provide users with marine data. The user can input an area, and get an output file with the values of certain marine variables of points within that area.

2.3 Related research

The widespread deployment of AIS systems and the accompanying abundance of data has led to interests from many different parties. Research has been done on the use of AIS data to improve safety of vessels and decrease offshore collisions and other undesirable events, in a field called *maritime situational awareness*. In Pallotta, Vespe, and Bryan (2013), AIS data is being used to extract routes and for anomaly detection in certain selected areas. Anomaly detection is seen as a deviation from normality as learned using historical data and can be useful in the detection of collision avoidance maneuvers. In (Kowalska and Peel, 2012) anomaly detection is done using Gaussian Process regression with active learning to indicate criminal activities, such as piracy, drug smuggling, arms trading, people trafficking and illegal immigration. Guillaume and Lerouvreur (2013) describes a model where clustering techniques are used to build a normalcy model to detect anomalies. In Bonham et

al. (2018) AIS data is used to classify ship behaviour using K-means clustering. The aim was to gain insights in ship behaviour to aid port traffic utilisation, port network analysis, and port delays.

The analysis of AIS data is applicable in the marine sector since the data is inherently connected to naval vessels, but the analysis of spatiotemporal data is being done in other fields too. GPS data is similar to AIS data in the sense that both data contain knowledge of the location and speeds of objects in time. In Alevizos, Artikis, and Paliouras (2017), Moosavi, Ramnath, and Nandi (2016) and Moosavi et al. (2017) GPS data of cars is used to segment trajectories and discover driving patterns, characterize driving behavior and forecast events using Markov chains. In Bijman (2017) features such as mean speed, maximum speed, acceleration and braking were used to classify driving behavior. Classes were found using clustering algorithms and drivers could be classified as safe or unsafe as a result.

We also looked for research on AIS data coupled with COPERNICUS data, but there appears to be very little research done on these subjects. The only one we were able to find was Goerlandt (2017). In this research, AIS is coupled with sea state data to model navigational accident scenarios in the Baltic Sea.

Author's note: Confidential research was omitted from this section

Chapter 3

Raw data

Author's note: This chapter is confidential

Chapter 4

Data enrichment

As discussed in the previous chapter, there are more variables we need. In this chapter we will describe the process of data enrichment, i.e. the process of taking the raw data and using that to find additional data that might prove useful in our project.

First, we will describe an enrichment based on the labeling of certain locations where ships find themselves. While this method can potentially tell us a lot about the activities of the ships, it is a very tedious process and takes a lot of time as the labeling is done by hand. Furthermore, as we can never be sure what is actually happening at those locations, the labeling can be seen as a subjective and dependent on the person doing the labeling.

Second, we will move on to a more objective way of enriching our data set that we will move on with for the rest of the thesis.

4.1 Location-based profiles

One way to define a user profile is based on predefined location and activity labels. First, we define all areas of interest and we assign to every area one of the following labels:

- port
- platform
- yard
- drop
- open

These location-labels have been agreed upon to be realistic and relevant after discussions with experts at Damen. See section 4.1.1 for a more detailed description of these locations.

From this, we get an overview of the activities of the ship. In the sections below, we will describe finding the areas of interest and their labeling.

4.1.1 Areas of interest

Finding and classifying ship locations were done by visually and by hand. The first step was to filter for data points with a speed higher than 1 knots. These points can be considered points at areas of interest because of their lack of movement in these locations. See figure 4.1 for an example of ship 2.

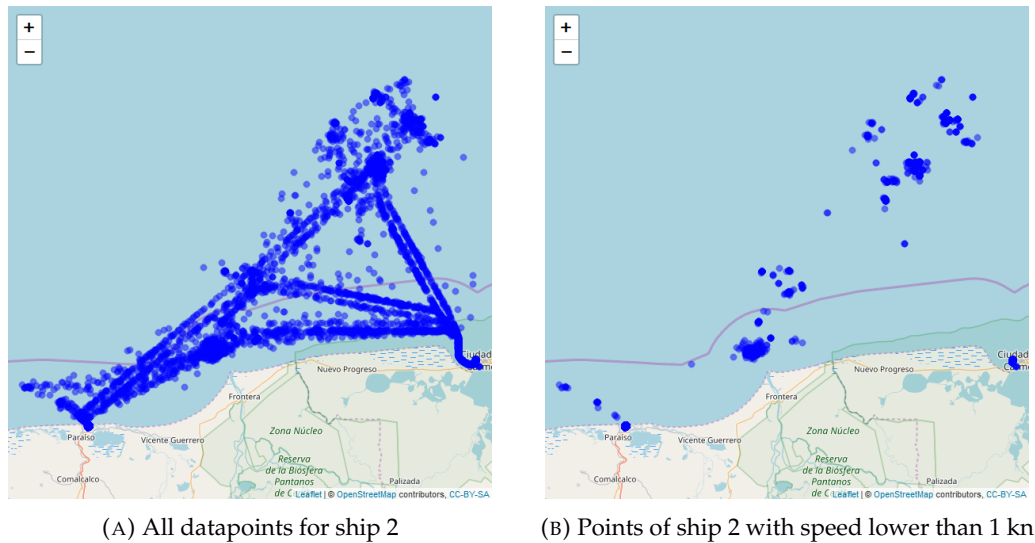


FIGURE 4.1: Points of ship 2

Although subjective, we can visually distinguish certain areas that indicate an area of interest for this ship. The next step is to label these areas according to the previously mentioned labels. A mix of GOOGLE MAPS and NAVIONICS was used for this. GOOGLE MAPS offers maps and satellite images of nearly all locations on earth and NAVIONICS offers an extensive map with different maritime facilities on it. Let us have a closer look at the labels to understand how we will label the areas.

Port

A port is a maritime facility with one or more wharves where ships can dock for a longer time. They are mostly recognizable on a map because they are connected to land, and there are wharves and docked ships visible.

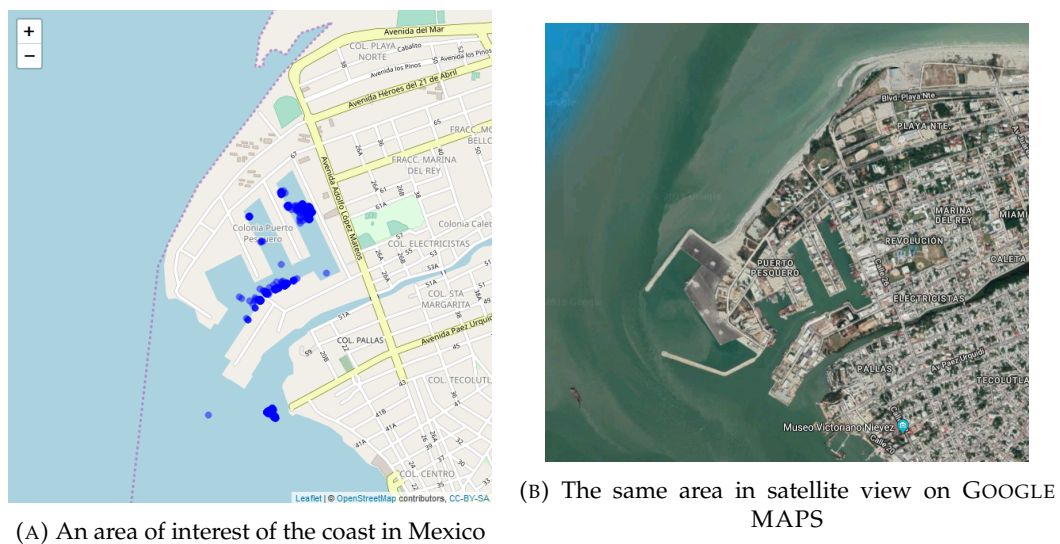
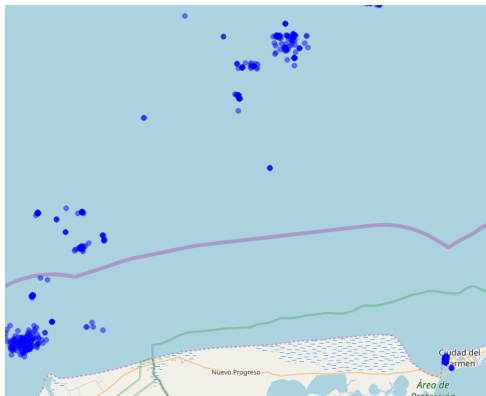


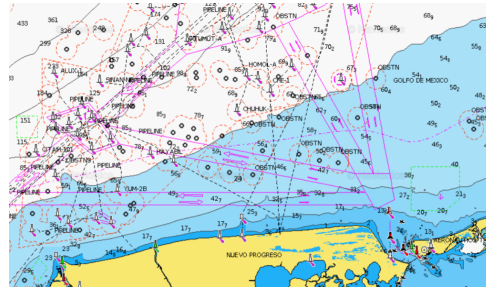
FIGURE 4.2: An area of interest compared to satellite images

Platform

The DAMEN FCS 5009 is being sold as a Fast Crew Supply for offshore operations, so most ships make trips to offshore platforms in their daily operations. Offshore platforms are typically not large enough to be seen by satellites, so determining their position by using GOOGLE MAPS is not possible. As we were not able to find a complete list of all platforms in the world, we chose to use the NAVIONICS webapp. See figure 4.3b.




(A) Coast off of Mexico with hot spots of datapoints



(B) The same area as 4.3a but on the NAVIONICS webapp

FIGURE 4.3: A comparison of the plotted data and a screenshot off NAVIONICS to find the positions of platforms

In figure 4.4 you can see a larger view of a part figure 4.3b where the individual platforms are distinguishable. Note that the  symbol stands for a platform.

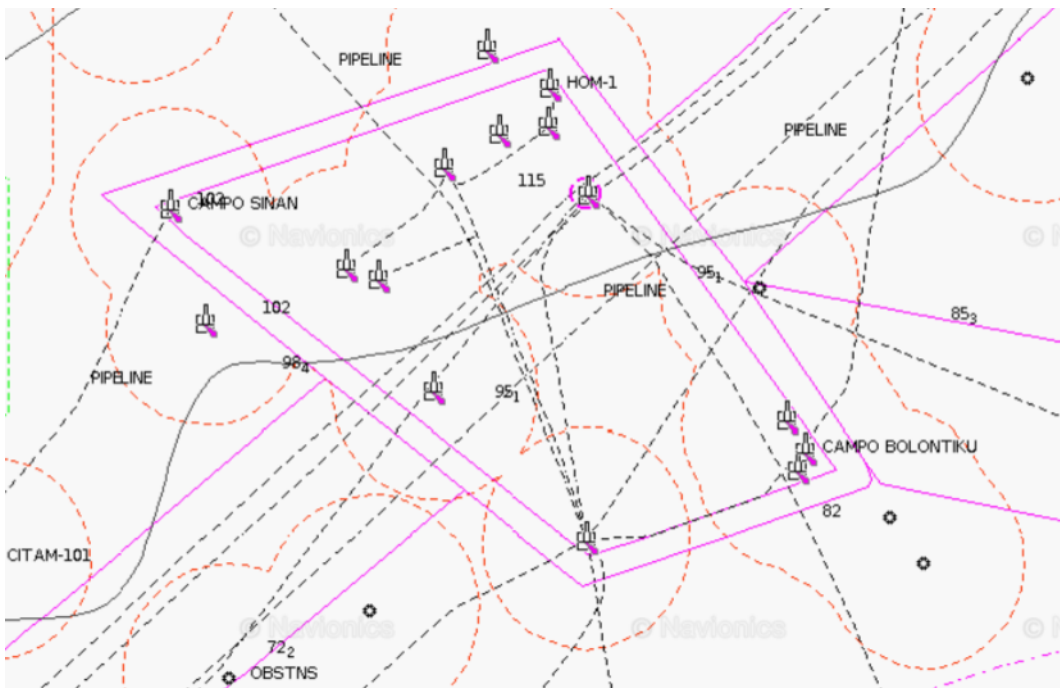
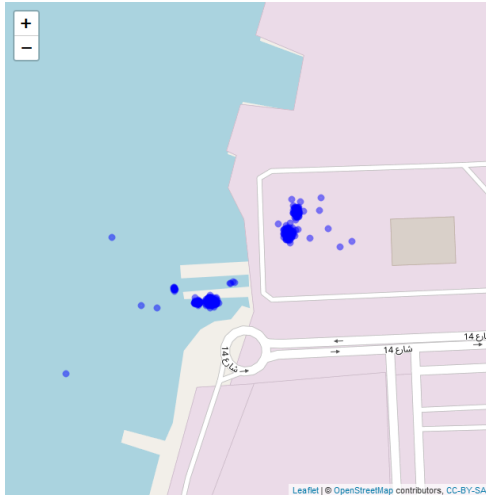


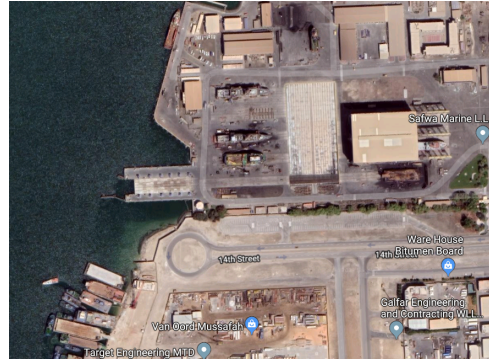
FIGURE 4.4: Zoomed in view of figure 4.3b

Yard

A yard is a maritime facility where ships are built or repaired. The easiest way to recognize them is when you see data points situated on land or when GOOGLE MAPS explicitly calls the location a yard. See figure 4.5.



(A) Datapoints off the coast of Abu Dhabi. Notice the points on land.



(B) The same area as 4.5a but on GOOGLE MAPS

FIGURE 4.5: A comparison of the plotted data and a screenshot off GOOGLE MAPS to find the positions of yards. Notice the ships on land in the satellite picture that indicates ships being built or repaired.

Drop

A small portion of the sold FCS 5009 are being used as yacht supports as explained earlier. These ships do not visit platforms, but rather islands where yacht owners can enjoy leisure activities. See figure 4.6 to see an example of a drop area.



(A) Datapoints off the coast of Papua New Guinea



(B) The same area as 4.6a but on GOOGLE MAPS

FIGURE 4.6: A comparison of the plotted data and a screenshot of GOOGLE MAPS to find the positions of drop areas

A drop area is named as such because of the assumed activities: it follows a yacht with passengers on board, and drops off any of the wanted vehicles and leisure ware and then waits until the yacht is ready to proceed the journey again. The areas are recognizable because of their locations close to land, but not at a port, and the fact that we know the ship is being used as a yacht support ship. However, note that ships being used as fast crew supply ships can still have drop areas, to drop off workers who will transfer to smaller ships headed to land.

Open

Open is a category of areas where we are not sure what the ships are doing, but we suspect there is something going on there. Basically whenever we see an area of interest where one of the previously mentioned labels do not apply, we label it 'open'.

So for any hot spot that we find when plotting the data on a map, we compare it to satellite images and NAVIONICS images to label the location.

4.1.2 Status labeling

When we have all the location labels of a given ship, we can start labeling the status of every datapoint that we have. A status can be seen as a mixture of both the location and the speed at a certain instant. From the previous subsection we have certain locations labeled as:

- port
- platform
- yard
- drop
- open

From these we get the following status labels:

- port
When a ship is in the port
- platform
When a ship is in an area labeled 'platform' and has a speed lower than 3 knots
- stand-by
When a ship is in an area labeled 'platform' and has a speed between 3 and 7 knots. Mostly observed in areas with a high platform density. Ships will patrol between platforms to provide quick assistance in case of emergency.
- yard
When a ship is in a yard.
- drop
When a ship is in a drop-area.
- open
When a ship is in an open-area.

- sailing
When a ship is in a location without any of the previously mentioned location-labels.

4.1.3 Critiques on labeling

As mentioned earlier, the above described way of labeling ship activities is very prone to erroneous judgement of the actual activity. The only reliable labeling is the labeling of ports and yards, since those are very easily distinguishable on satellite images. The distinction between platform, drop and open however is not always easy to be seen. One issue with this is that there was no way to pinpoint the exact latitude/longitude coordinates of platforms on NAVIONICS, so matching locations was mainly done by matching certain landmarks to orientate. Another problem is that oil- and drilling rigs do not always stay in one place, but rather get moved around from time to time. So when the location of an area of interest does not coincide with a platform on NAVIONICS, it does not necessarily mean that there was no platform there at the time when the ship was there. This was not considered at the beginning of the process, but was discovered as the process went on.

But consequently, this led to questioning of the labeling of locations thus far. We felt the classifications that were done were not very accurate and continuing with this would be going the down wrong path. After discussions with people within Damen, it was decided to focus on profiles based on speed, rather than locations. We managed to label the locations of 24 ships before coming to this decision.

Before we move on to our next section, we do want to propose a method of automatically finding areas of interest. Finding areas of interest, or hot spots, is a big topic in spatial statistics. Applications include industries as health care, nature observation, transportation, and many more. In Harris et al. (2017), hot spots analysis is used to identify hot spots of emerging forest loss. In Nandana, Mala, and Rawat (2019) hot spots detection is used to analyse dengue fever outbreaks. In Sitanggang, Risal, and Syaufina (2018) hot spots are identified to analyze fires in the country, and in Qin et al. (2017) it is used to analyze taxi movements in the city. In the following section we will describe a method called *DBSCAN*.

4.2 Finding areas of interest based on location density

To gain insight into the activities of ships and their behavior, one piece of information that is of interest is where they go outside of the port. The place they go to are what we call the areas of interest. We show again an image of all the points of a certain ship in off the coast of Mexico from a 3-year period as in figure 4.1a, but this time with the areas of interest circled as one might suspect by looking at the densities of the data and the paths leading there. See figure 4.7.

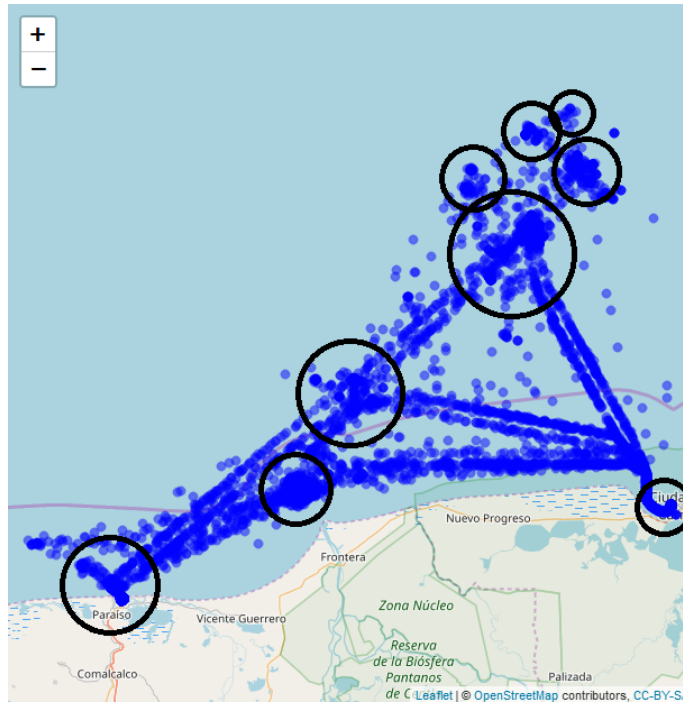


FIGURE 4.7: All datapoints from ship 2 with visually distinguishable areas of interest circled

By only plotting the points where the speed was lower than 1 knot as in figure 4.1b, we see that the data points fall mostly into the circles of figure 4.7. We would however want to find these areas automatically, and not by hand. That is what this section is about. We use an algorithm called DBSCAN to achieve this. The algorithm is first described in Ester et al. (1996) and in Yu et al. (2014) the effectiveness of the algorithm in geospatial hot spot exploration is described.

DBSCAN stands for "Density-Based Spatial Clustering of Applications with Noise" and as the name implies, is a density-based clustering algorithm.

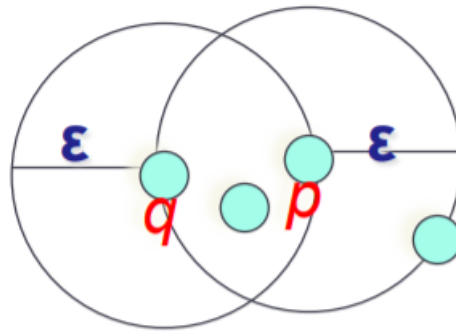
In the following section we explain the DBSCAN algorithm as described in Ester et al., 1996.

4.2.1 DBSCAN

Let D be a data set of points $p \in \mathbb{R}^n$. To perform DBSCAN on the data D you need to specify two parameters:

- ϵ : the radius for the ϵ -neighbourhood N_ϵ of p .
 $N_\epsilon(p) = \{q \text{ in } D \mid \text{dist}(p, q) \leq \epsilon\}$
 where $\text{dist}()$ is a distance function (often Euclidian, but not necessarily)
- minPts : the minimum number of points needed in a neighborhood

We illustrate this with figure 4.8. In the figure we see four datapoints \mathbb{R}^2 , in blue. Around the points p and q are the circles that represent the ϵ -neighborhoods of those points with radius ϵ .

FIGURE 4.8: Illustration of N_ϵ

With these notions, we can label all points as one of the following points:

- **Core Point:** Points of which their neighborhood contain at least $minPts$ points
- **Border Point:** Points of which their neighborhood contain less than $minPts$ points, but they are in the neighborhood of another core point
- **Noise Point:** Points that are neither core nor border points

Looking back at figure 4.8, we can label points p and q . Let us say that $minPts = 4$. Then $N_\epsilon(p)$ contains 4 points and is therefore a core point. $N_\epsilon(q)$ however only contains 3 points and is therefore not a core point. It is however located in the neighborhood of core point p , and is therefore a border point.

To form the clusters, we have some other notions to define.

Directly density-reachable

A point q is directly density-reachable from a point p if:

- p is a core point, and
- $q \in N_\epsilon(p)$

Density-reachable

A point q is density-reachable from point p if there is a sequence of points p_1, p_2, \dots, p_n with $p_1 = p$ and $p_n = q$ such that p_i is directly density-reachable from p_{i-1} for $i = 2, \dots, n$.

Density-connected

Two points p and q are density-connected if they are both density-reachable from a core point $k \in D$.

Given the definitions above, we can define a cluster:

Cluster

A cluster C is a subset of D satisfying the following two criteria:

- **Maximality**
 - $\forall p, q$ if $p \in C$ and q is density-reachable from p , then $q \in C$
- **Connectivity**
 - $\forall p, q \in C$, p and q are density-connected

In the DBSCAN algorithm, we start by finding a core point, and then find all points in D that are density-reachable from p and assign all these points into a new cluster. Then move on to a new core point that has not yet been labeled and repeat the process. See algorithm 1 for the pseudocode.

Algorithm 1 Pseudocode DBSCAN

 $\forall p \in D$

- 1: **if** p is not classified **then**
 - 2: **if** p is core point **then**
 - 3: Find all points density-reachable from p and assign them to a new cluster
 - 4: **else**
 - 5: assign p as noise
-

Author's note: This section is confidential.

In this section we have shown that the DBSCAN algorithm performs well at finding hot spots, or areas where ships stand still. Finding these spots can be useful for analysis, as they give insight in what the ships are being used for. However, simply using this algorithm is not enough. Domain knowledge is still necessary to label these hot spots (e.g. platform, mooring buoy, port, yard). Since we deemed it not possible for us to adequately do this, we will not continue with these labelings in the remainder of this study. However, we did report on this process in this report since we think that the idea of labeling locations and activities, if done correctly, can be very useful for Damen. With the DBSCAN algorithm that we explored here we have shown that should Damen choose to find and label hot spots, the algorithm could help them in the process.

4.3 Speed-based profiles

We described a way to enrich the data with location-based features in the previous section, but it proved unreliable. We therefore move on to another, more reliable way, which is speed-based. A big factor that plays into a ship's life expectancy is the speed the ships sail with and the duration. Therefore it would make sense to make clusters based on speed. The first step in this however is still to label all ports and yards like we did before. We need to do this because Damen is interested in the behaviour of the ship when it is sailing. It would therefore provide more information to know both how often the ship is in or out of the port, and the behaviour when it is not in port.

4.3.1 Division into trips

Like before, we identify all ports that a ship visits in the data set that we have. For all 46 ships, we identified 141 ports. See figure 4.9.

Author's note: This figure is confidential

FIGURE 4.9: All identified ports in the dataset

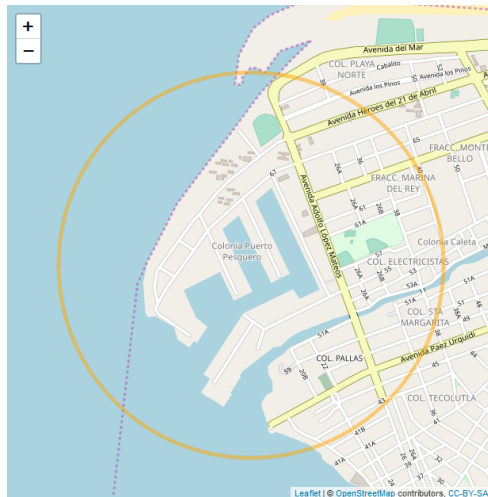
We identified these ports by analyzing the hot spots found with DBSCAN. When a port was found, we saved one longitude/latitude pair and a radius that captured the port. Note that this includes ports from small (artificial) islands, for example one that can be seen in figure 4.10.



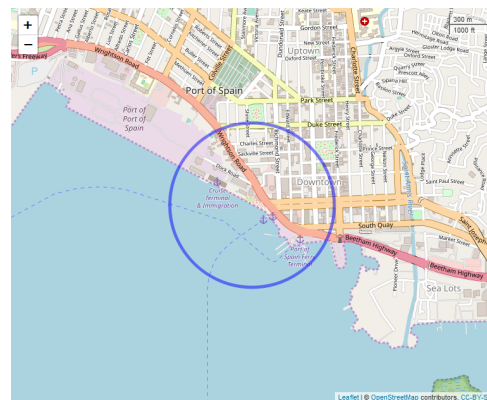
FIGURE 4.10: Small artificial island in the Persian Gulf that we also identified as a port.

From this, we can calculate the distance from this point to any location of the ship. If the distance is smaller than the radius, we label the location of the ship as the

port location. See figure 4.11. As one can see in the figure, it is not perfect. There are points outside of the port that would still be labeled as the port in this instance, but it would take too much time to draw a perfect outline of all 141 ports. Furthermore, it is not always clear where ships can dock at a port. From figure 4.11a one might think all ships will be stored in the lanes in the middle of the circle, but figure 4.2 shows that ships can also get stalled just outside of what seems like the whole port. So when a location of a ship gets labeled as in a port, we will keep in mind that it can also mean that the ship is at least very close to a port. We do not expect this to cause any issues further down the line.



(A) Port Pesquero in Mexico with the radius that we saved. In our labeling, we consider every point within the circle as "in Port Pesquero".



(B) Port of Spain in Trinidad and Tobago with the radius that we saved. In our labeling, we consider every point within the circle as "in Port of Spain".

FIGURE 4.11: Two examples of ports and how we label points as in these ports.

With all the ports identified, we know of every data point whether it is in port or not. From this we can define trips:

Trip

A consecutive sequence of points of which:

- The first point in the sequence is located at a port
- All points from the second until the second to last one are located outside a port
- The last point in the sequence is located at a port

In this way the full sequence of data points per ship gets divided in sequences in port and sequences away from port.

Furthermore, now that we know which points are located in a port and which are not, we can again plot the figures that we looked at in the raw data analysis. This time however we can take out all points located in a port. We do not think these plots fit in well in this chapter however, so we refer the reader to appendix C for the plots and comparison.

4.3.2 Estimating average speeds

As mentioned in chapter 3, the resolution of our data is roughly one per hour. For all these data points, we have a point estimate of the speed calculated by the GPS system on board at the ship. We are however more interested in the average speed in between two data points, and this section will describe a method to estimate this average speed. More precisely, see figure 4.12.

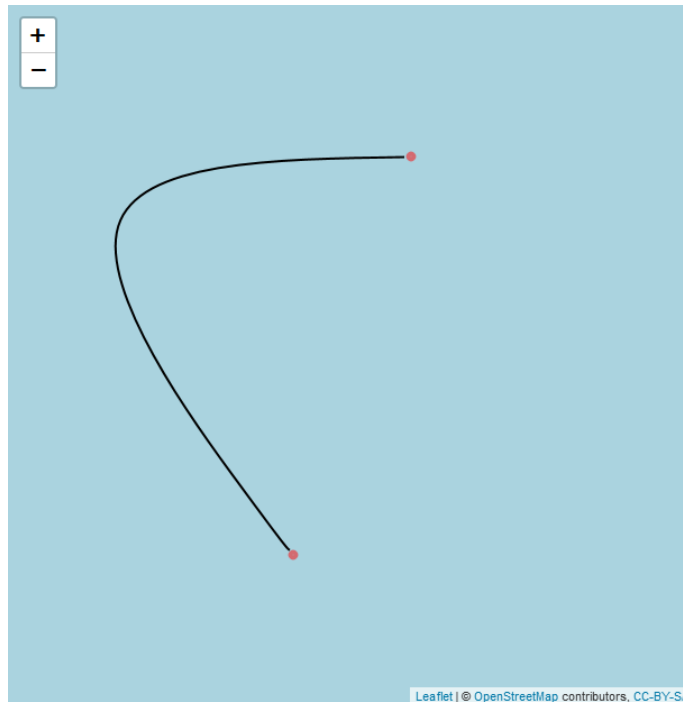


FIGURE 4.12: Location of two data points and the sailed path that we want to estimate.

The two red dots are locations of two data points that we have, and the black line is the real path that the ship sailed in between these points. Note this is just an example, we do not actually know the real path. To get the average speed sailed on this path, we need to know the length of this path and the time elapsed during this path. Let $L(t) = (x(t), y(t))$ denote the positions of our points in time. The average speed between $L(t_1)$ and $L(t_2)$ can then be calculated as

$$\frac{\int_{t_1}^{t_2} \sqrt{\dot{x}(t)^2 + \dot{y}(t)^2} dt}{t_2 - t_1}$$

However, we can not do this if we only know $L(t_1)$ and $L(t_2)$. In the following we will describe a method to get an estimate of this path. We were able to attain a more detailed data set of four of our ships, with a resolution of roughly 6 data points per hour. We will use this data set to tune some parameters of our method and to see how well our method performs.

Interpolation

This section describes an interpolation between points to get a (smooth) curve between the points and enrich the data with the distance traveled between points and

get the average speed between two points. The first decision we need to make when trying to obtain smooth curves from data, is whether we want to interpolate or approximate. Interpolation means finding a line passing through all given points, while an approximation does not necessarily do so. See figure 4.13 for an illustration.

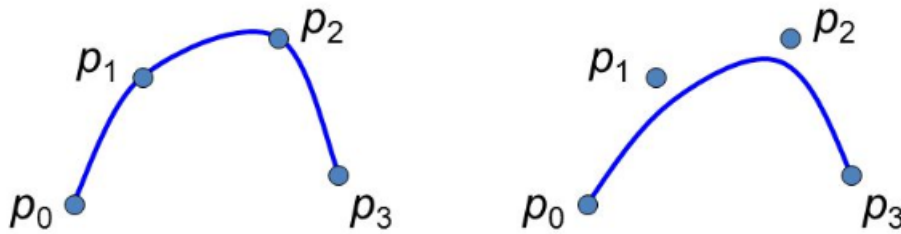


FIGURE 4.13: Four points, with an interpolation on the left and an approximation on the right

First we removed inconsistencies in the data. Inconsistencies are locations that are physically unreachable from the previous location given the time in between two data points, or points with impossible sailing speeds. After removing the inconsistencies, we assume the data, e.g. the positions, to be accurate. Approximating a curve between the data points, and thereby a path taken by the ships, would potentially miss out on locations the ships visited. Especially since we only have the data once per hour (at best), approximating the path does not seem suitable. We therefore will opt for interpolating methods. Our goal is to estimate a smooth path the ship has taken between two consecutive points. Our estimation should satisfy two properties:

- The estimation passes through all the points
- The tangent of the estimation is equal to the course of the ship at every data-point with a significant speed.

Bézier curves are smooth curves that are defined by a set of control points. The curve always passes through the first and last control point, but the intermediate control points generally do not lie on the curve. They do however influence the curve. We will show some examples and then make it more definitive.

In figure 4.14 you can see two Bézier curves with four control points. In both examples, the curve passes through the first and last points P_0 and P_3 but not the intermediate points P_1 and P_2 . It is however clear that the path of the curve is guided by the intermediate points.

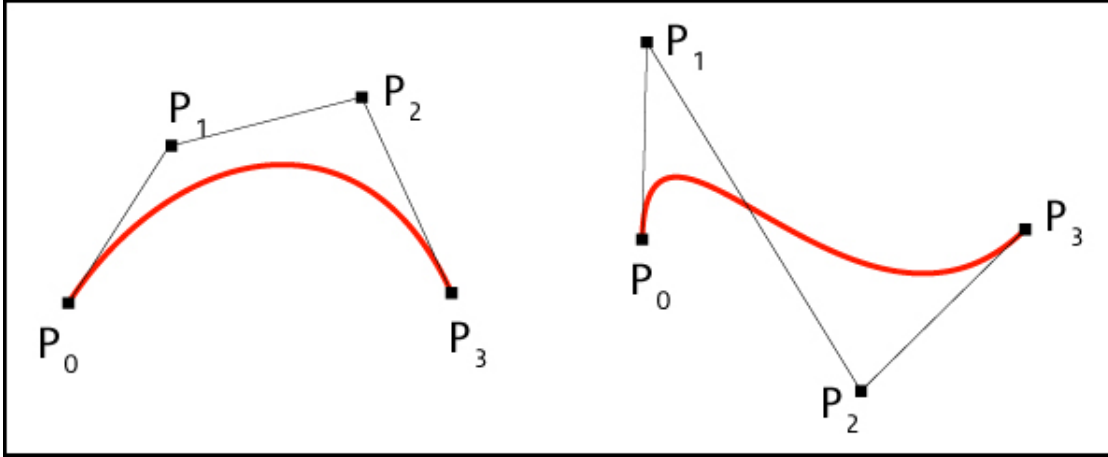


FIGURE 4.14: Two Bézier curves with four control points, in different configurations

So by moving the control points, one can change the curve to fit many shapes. The ease and seemingly intuitive way of shaping the curve makes it a widely used method in computer graphics and computer-aided design. In the next section we will describe the Bézier method in more detail.

4.3.3 Bézier

In this section we will explain the Bézier curve as can be found in Marsh and Marshall (1999). **Definition Bézier curves** The Bézier curve in a plane is a parametric curve of the form $B(t) = (x(t), y(t))$, where B is the curve, t is the parameter and $x(t)$, $y(t)$ are the polynomial coordinate functions. The degree of the curve is the highest order of the parameter in any coordinate function. Furthermore, given the $n + 1$ so-called control points $b_0, b_1, \dots, b_n \in \mathbb{R}^2$, the Bézier curve of degree n is defined to be

$$B(t) = \sum_{i=0}^n b_i B_{i,n}(t) \quad (4.1)$$

where

$$B_{i,n}(t) = \begin{cases} \frac{n!}{(n-i)!i!} (1-t)^{n-i} t^i & \text{if } 0 \leq i \leq n \\ 0 & \text{otherwise} \end{cases} \quad (4.2)$$

are the so-called Bernstein polynomials or Bernstein basis functions of degree n .

So when we express the control points as $b_i = (x_i, y_i)$ for $i = 0, \dots, n$, we have that

$$B(t) = (x(t), y(t)) = \left(\sum_{i=0}^n x_i B_{i,n}(t), \sum_{i=0}^n y_i B_{i,n}(t) \right)$$

Properties of Bézier curves

A Bézier curve of degree n with control points $\mathbf{b}_0, \mathbf{b}_1, \dots, \mathbf{b}_n$ for $t \in [0, 1]$ has the following properties:

Endpoint Interpolation Property: $B(0) = \mathbf{b}_0$ and $B(1) = \mathbf{b}_n$

Endpoint Tangent Property: $B'(0) = n(\mathbf{b}_1 - \mathbf{b}_0)$ and $B'(1) = n(\mathbf{b}_n - \mathbf{b}_{n-1})$

Convex Hull Property : For all $t \in [0, 1]$, the curve $B(t)$ lies within the convex hull of all the control points that define $B(t)$.

Invariance under Affine Transformations: Let T be an affine transformation, then

$$T \left(\sum_{i=0}^n \mathbf{b}_i B_{i,n}(t) \right) = \sum_{i=0}^n T(\mathbf{b}_i) B_{i,n}(t)$$

Variation Diminishing Property: For a Bézier curve $B(t)$ in a plane this property states that the number of intersections of a given line with $B(t)$ is less than or equal to the number of intersections of that line with the control polygon.

The endpoint tangent property states that the start and end of the curve coincide with the first and last control point, as we have seen in the examples.

The second property states that the line from the first to the second control point lies tangent to the Bézier curve in $B(0)$ and the line from the second to last control point to the last control point lies tangent to the Bézier curve in $B(1)$.

See appendix **D** for proofs.

4.3.4 Low degree Bézier curves

Whilst the results we got above is a general result true for all n , it is common to have no more than 4 control points, making the degree $n = 3$. The same is true for our case, which will be showed later. First let us examine the Bézier curves of degree up to 3.

Linear Bézier Curves

A Bézier curve defined by only 2 control points $\mathbf{b}_0, \mathbf{b}_1$ is just a line segment connecting the two control points and is given by

$$B(t) = \mathbf{b}_0(1 - t) + \mathbf{b}_1 t \quad \text{for } t \in [0, 1]$$

Quadratic Bézier Curves

Three control points $\mathbf{b}_0, \mathbf{b}_1, \mathbf{b}_2$ result in a Bézier curve of the second order with the following quadratic Bernstein functions that can be seen in figure **4.15**:

$$\begin{aligned} B_{0,2}(t) &= (1 - t)^2 \\ B_{1,2}(t) &= 2(1 - t)t \\ B_{2,2}(t) &= t^2 \end{aligned}$$

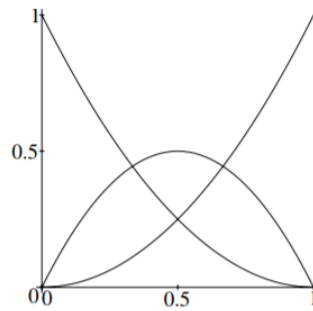


FIGURE 4.15: Parabolic Bernstein polynomials

and we get

$$B(t) = b_0(1-t)^2 + 2b_1(1-t)t + b_2t^2 \quad \text{for } t \in [0, 1]$$

Cubic Bézier Curves

A Bézier curve defined by four control points b_0, b_1, b_2, b_3 result in a curve of order 3 with the following cubic Bernstein polynomials that can be seen in figure 4.16 :

$$B_{0,3}(t) = (1-t)^3$$

$$B_{1,3}(t) = 3(1-t)^2t$$

$$B_{2,3}(t) = 3(1-t)t^2$$

$$B_{3,3}(t) = t^3$$

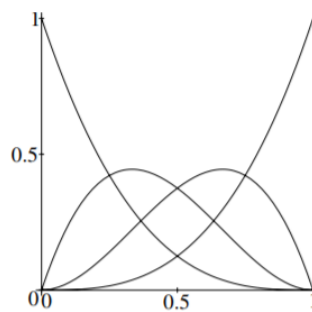


FIGURE 4.16: Cubic Bernstein polynomials

that lead to the Bézier curve:

$$B(t) = b_0(1-t)^3 + 3b_1(1-t)^2t + 3b_2(1-t)t^2 + b_3t^3 \quad \text{for } t \in [0, 1]$$

4.3.5 Application Bézier

In this section we show the Bézier curves we get when applying the describe method to our data. As described earlier, we want to interpolate our data to estimate a path the ship has taken between two consecutive points. As said in the previous section, we want the following satisfied:

- The estimation passes through all the points

- The tangent of the estimation is equal to the course of the ship at every data-point with a significant speed.

We can achieve this, by adding control points in the direction of the course of the ship at every point. In this way, we are actually using the Bézier method, which is an approximating method, to interpolate our data. First, we use the data to create a set of pairs of the form $\{(\mathbf{d}_1, \mathbf{d}_2), (\mathbf{d}_2, \mathbf{d}_3), \dots, (\mathbf{d}_{n-1}, \mathbf{d}_n)\}$ where the vector \mathbf{d}_i is one data point with the values of the variables described earlier (latitude, longitude, speed, et cetera) for $i = 1, \dots, n$. Then we can interpolate between the points \mathbf{d}_i and \mathbf{d}_{i+1} by adding control points between \mathbf{d}_i and \mathbf{d}_{i+1} . Using the Bézier method as described above on these four points will then result in a curve of degree 3. The location of the two control points are determined by \mathbf{d}_i and \mathbf{d}_{i+1} .

See figure 4.17 for an illustration of the interpolation between two points. This is done for the first point in the direction of the course of the ship at that point. Let (lng_i, lat_i) , ϕ_i and v_i be the coordinates, course and speed of \mathbf{d}_i . Then the coordinates of the of the first control point are calculated as

$$lng_{c_i} = lng_i + \frac{v_i \cdot \cos\left(\frac{\phi_i}{180}\pi\right)}{scale}$$

$$lat_{c_i} = lat_i + v_i \cdot \frac{\sin\left(\frac{\phi_i}{180}\pi\right)}{scale}$$

The coordinates of the second control point are calculated in a similar way, except i changes in $i + 1$ and the second term being subtracted instead of added. The scale term is needed to get a plausible estimation of the path. After experimentation the scale is set to 300. For the computation of the curve, discretization is needed. 30 steps is chosen as it results in fast computation times while still giving smooth curves as seen in figure 4.17c.

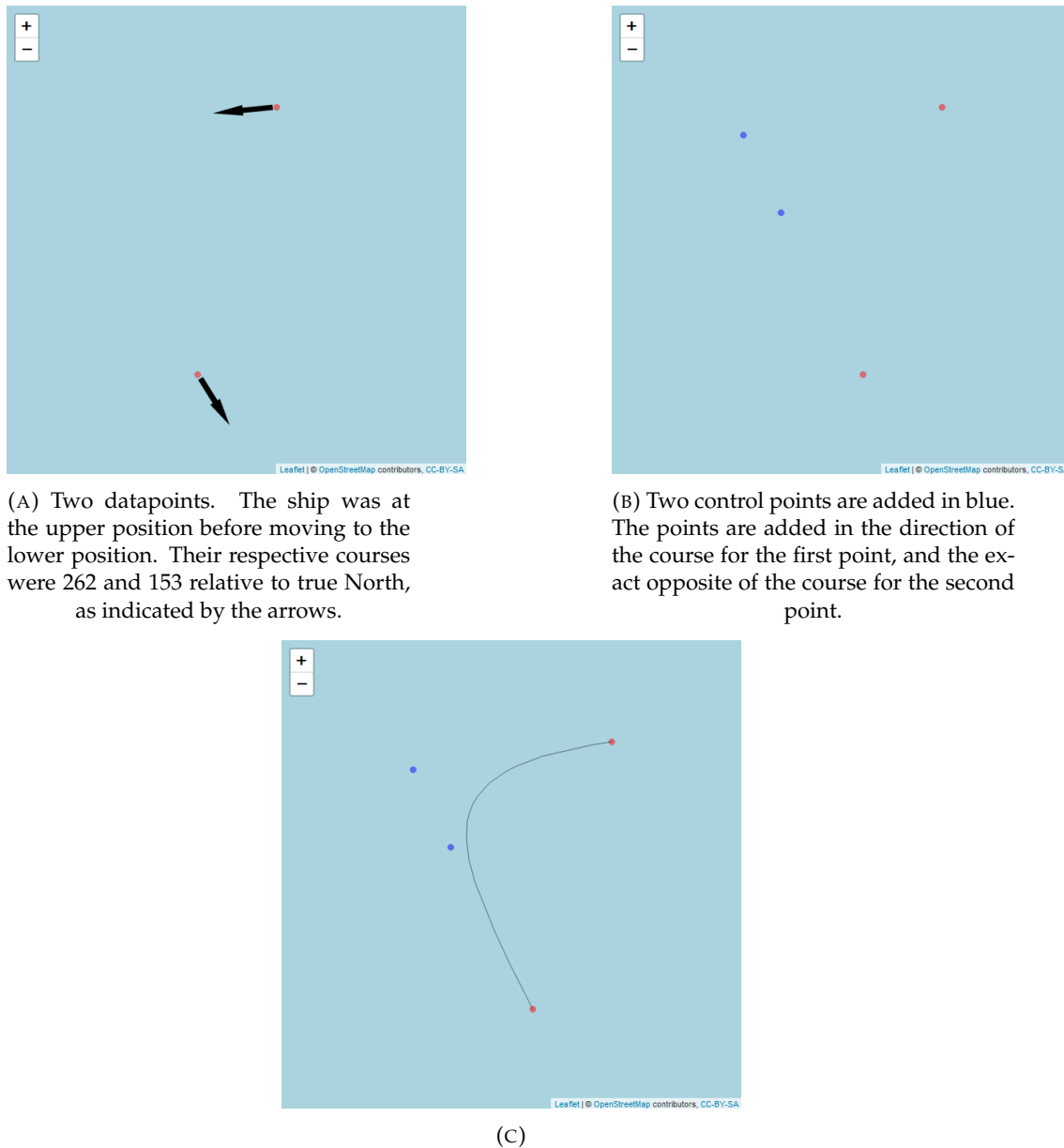


FIGURE 4.17: The resulting Bézier curve using 30 discretization steps.

4.4 Results Interpolation

In this section we investigate the results of our Bézier curves and how close they are to the real path. For this, we use a data set that was made available in the middle stage of this research project. This data set was purchased due to interest from another group within the research department of Damen, but it is also useful for our research. This data set consists of AIS data of four of our ships, but the data frequency is higher. Where our original data set contains one data point every hour, this high frequency data set has one data point roughly every 10 minutes. Note that the data from the same four ships that we started this research with, is a subset of this high frequent data. In the rest of this section we use the term non-detailed data to refer to data taken from our initial data set. The term detailed data refers to data taken from this new high frequency data set.

We use the detailed data to test the validity of our interpolation. First take a look at figure 4.18.

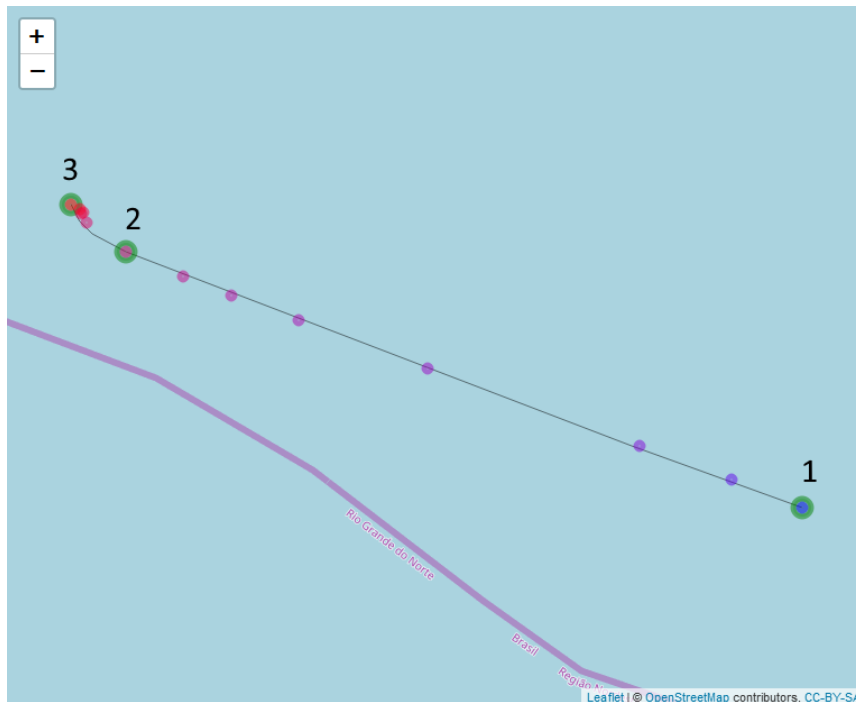


FIGURE 4.18: An overlay of our data with a Bézier curve and the detailed data.

In this figure we see data from both our non-detailed data set together with data from the detailed data set. The bigger green points 1, 2, and 3 are points from the non-detailed data set, and the curve that connects them is the Bézier interpolation, calculated as described in the previous section. The coloured points in between are points from the detailed data set, showing the locations of the ship at intervals from 5 to 17 minutes. So note that a greater distance between points does not necessarily imply a higher speed, it could also be due to a larger time interval. The colour shifts from blue to red, indicating the first to last point of the data. So in figure 4.18, the ship sailed from the right hand side to the left hand side.

We started this section with the wish to estimate the average speed in between two data points. We described the Bézier method, and we will compare it with two more straightforward methods. See below the three methods:

- Calculating the distance over the Bézier curve and divide that by the time between points 1 and 2.
- Calculate the shortest distance between 1 and 2 and divide that by the time.
- Take the speed at position 1 as the average.

To get a sense of how well a certain method performs, we would like to compare it to the real average speed. This is however obviously not possible. We do have the detailed data however, from which we think we can get a good estimate of the real average speed. We assume the intervals of roughly 10 minutes are short enough to contain most path information. Then if we take the shortest distance in between the detailed points and sum over those distances, we get again an estimate of the real path. Dividing that by the time duration of the interval, we get an average speed

estimate. From our assumption that the ship does not deviate much from the direct paths in between the short interval, this average speed should be quite close to the real average speed. We therefore take this average speed as reference, and compare the three aforementioned speeds to this speed. We choose not to use the Bézier interpolation on the detailed data, since our method is tuned to the intervals of roughly one hour. Therefore the method would not work on the detailed data. However, as said before, we assume that the ship does not deviate much from the direct path as calculated from the detailed data, so the estimate is assumed to be close to the real average speed.

We will call our first method *Bézier*, the second method *Haversine* (named after the Haversine formula for calculating the shortest great-circle-distance given two coordinate locations), and the third method *simple*. We repeat this for the interval between points 2 and 3, and show the results in figure 4.19. In this figure we can see that the Bézier and Haversine methods do not differ that much from each other, but both are closer to the real data than the simple method.

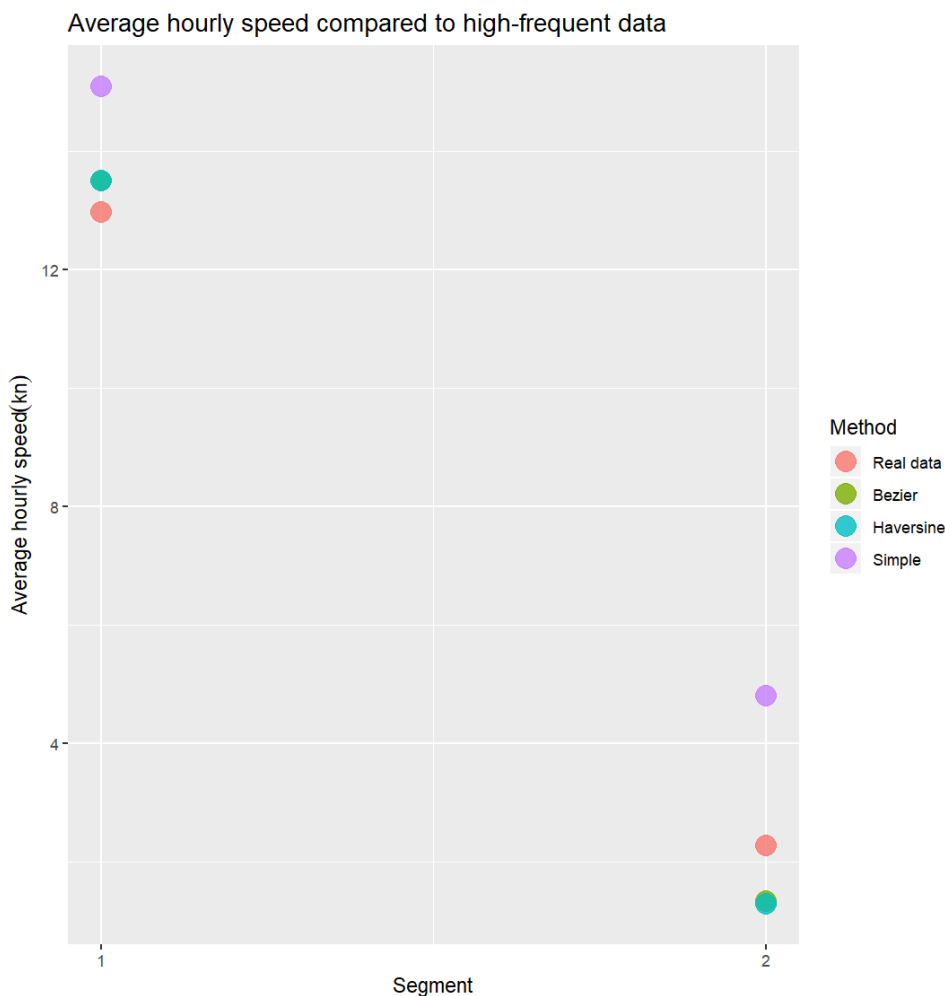


FIGURE 4.19: A comparison of the average speed between the two segments using the different methods and the detailed data. By segment 1 we mean the path between points 1 and 2 in figure 4.18, and by segment 2 we mean the path between points 2 and 3.

If we use more points than 3, a figure like figure 4.19 would become messy. So

we choose to look at the total squared error, which is cumulative sum of the squared distance from the real data average to the calculated average using the three methods described above. See figure 4.20 to see the total squared error in between the three points that we looked at earlier.

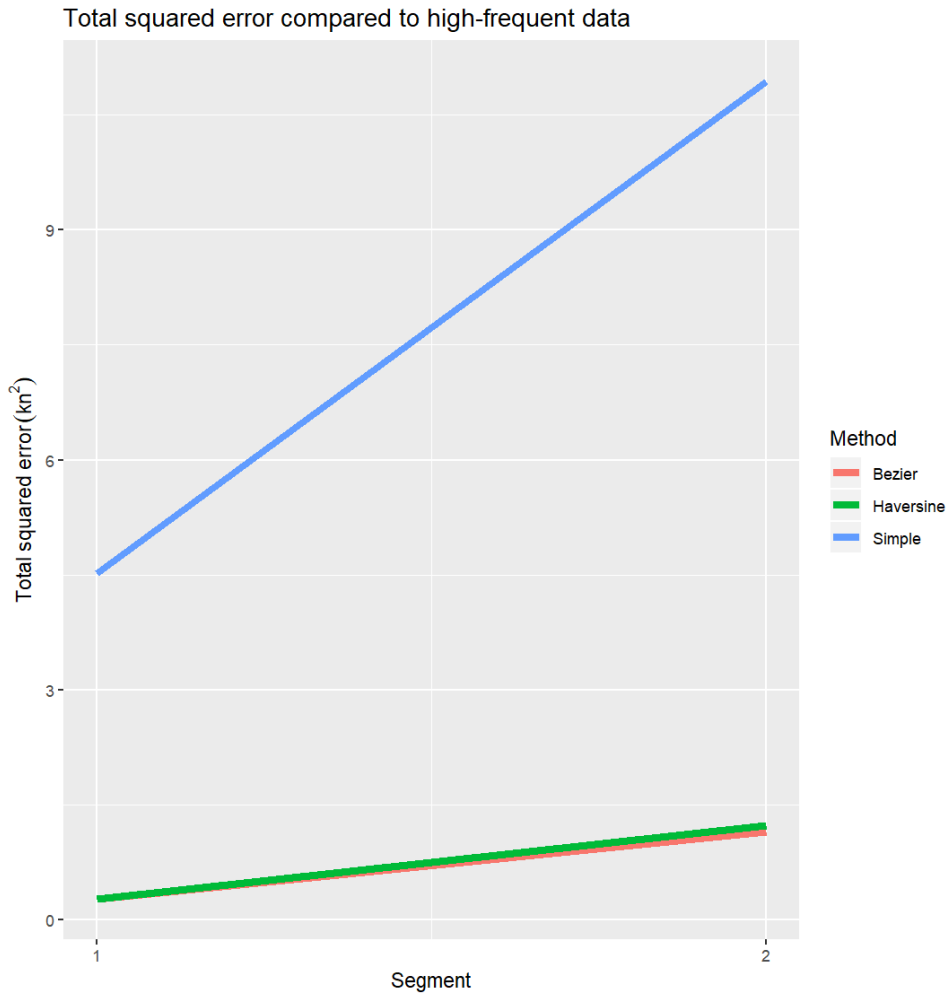


FIGURE 4.20: The total squared error from the detailed data to the estimating methods.

We do this for the whole non-detailed data set instead of just the three points and get figure 4.21.

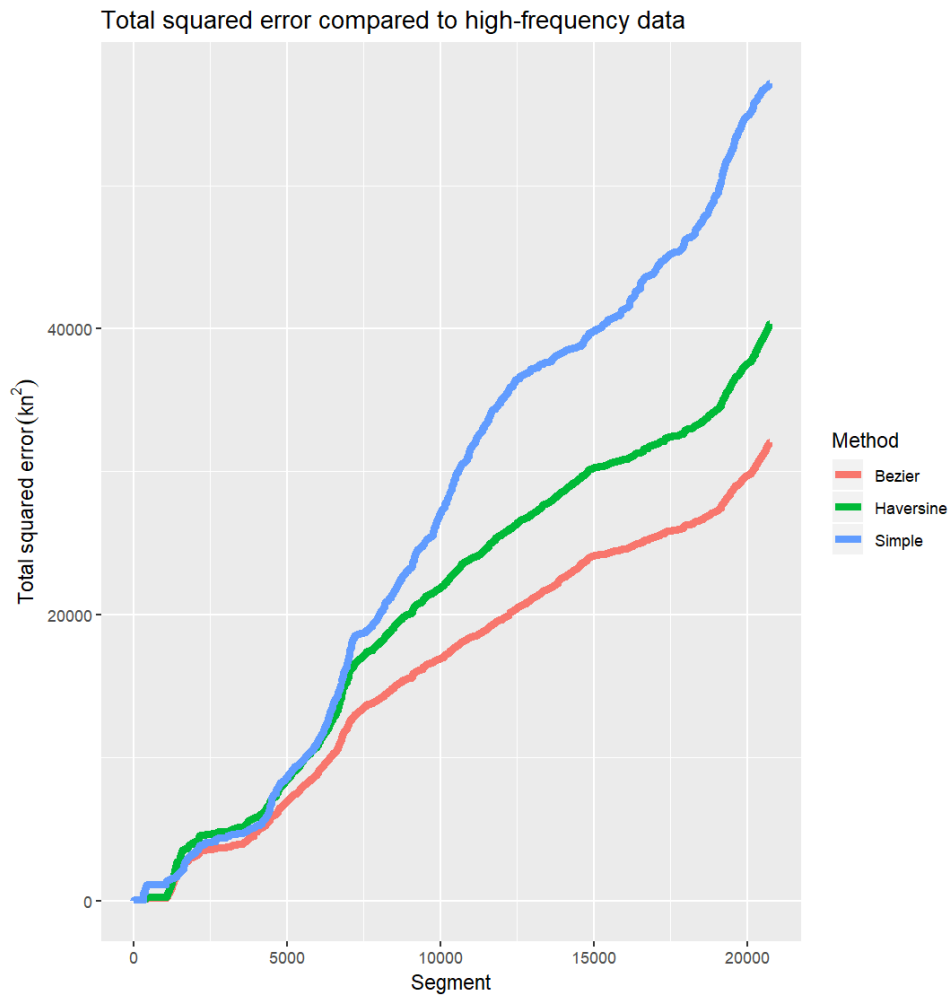


FIGURE 4.21: The total squared error from the detailed data to the estimating methods over the whole non-detailed set.

From this figure we can see that the Bézier methods leads to the lowest total squared error. The differences however mainly seem to start appearing about a third way through. In the beginning we can see that the simple method performs just as well, and at times even better.

The differences are somewhat explainable by looking at different parts of ship's journey. See the sailing pattern of the ship in a certain period in figure 4.22. The ship does not seem to have a clear destination, it is rather sailing back and forth in this area.

Author's note: This figure is confidential

FIGURE 4.22: A certain sailing pattern of ship 40 off the coast of Brazil.

Let us again take 3 selected points out of this to see what the effects will be. See figure [4.23](#).

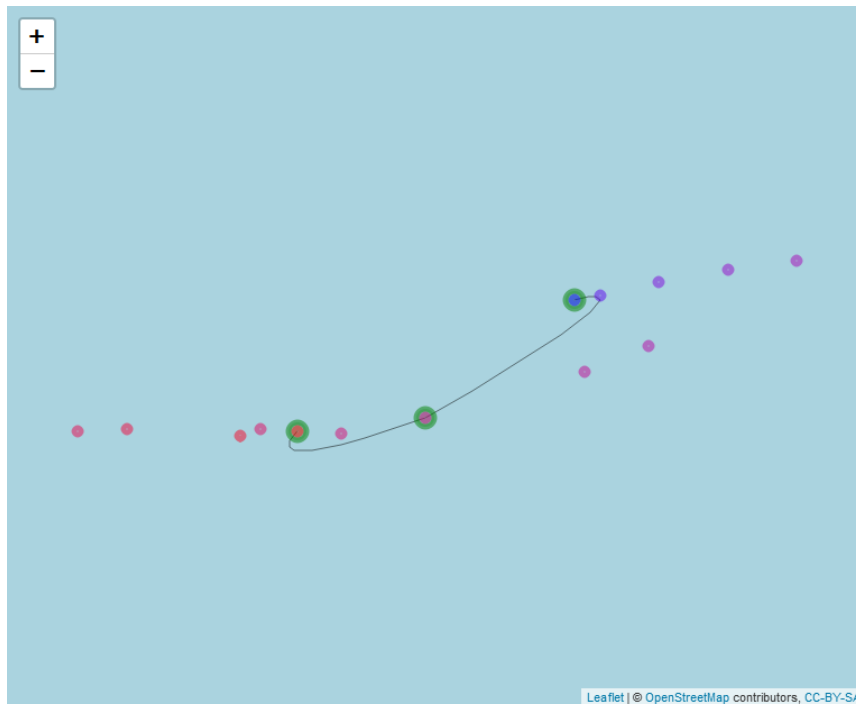


FIGURE 4.23: Three points from our non-detailed data set overlaid with the detailed set in the area where the ship is sailing back and forth.

The speeds are quite low, around 3 knots, and therefore the Bézier method does not extend too far out. What we see here is a form of aliasing, where low frequency sampling of patterns with high variation leads to erroneous results. See figures 4.24 and 4.25, where we see that in such cases, the simple method performs better.

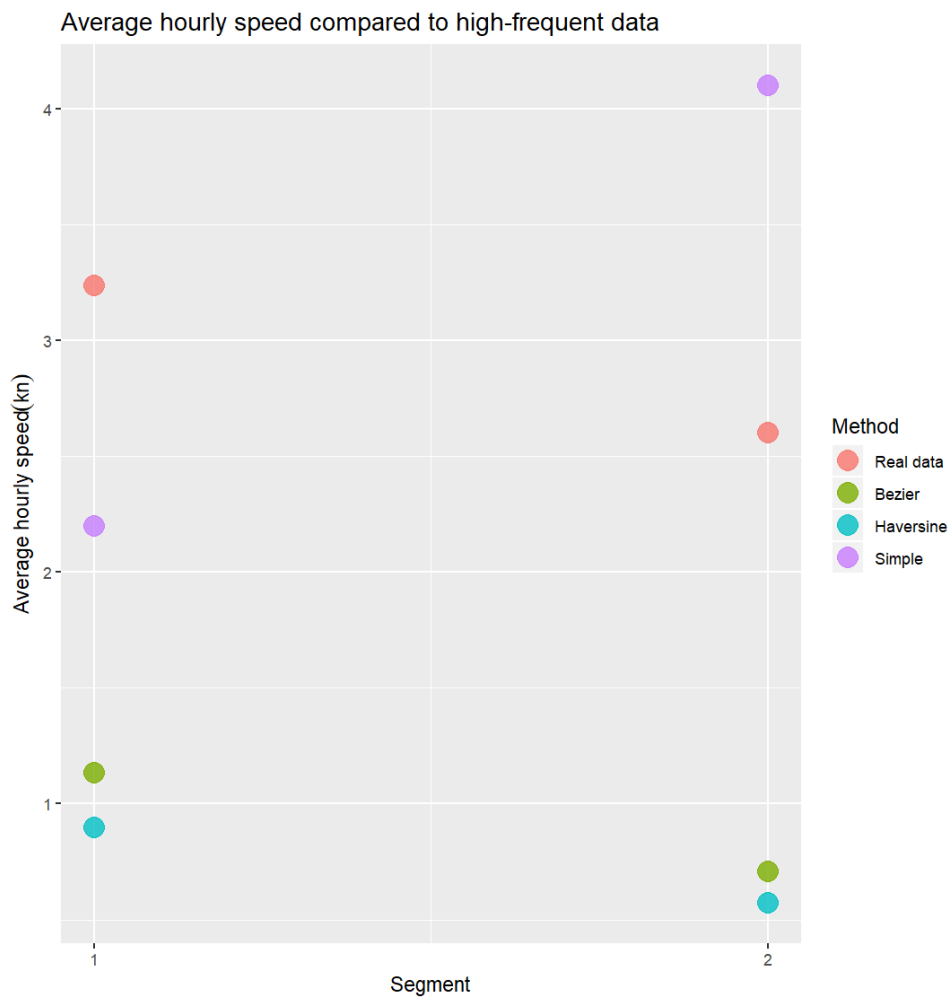


FIGURE 4.24: The average speed between the three points in figure 4.23 calculated using the three different methods and calculated from the detailed data.

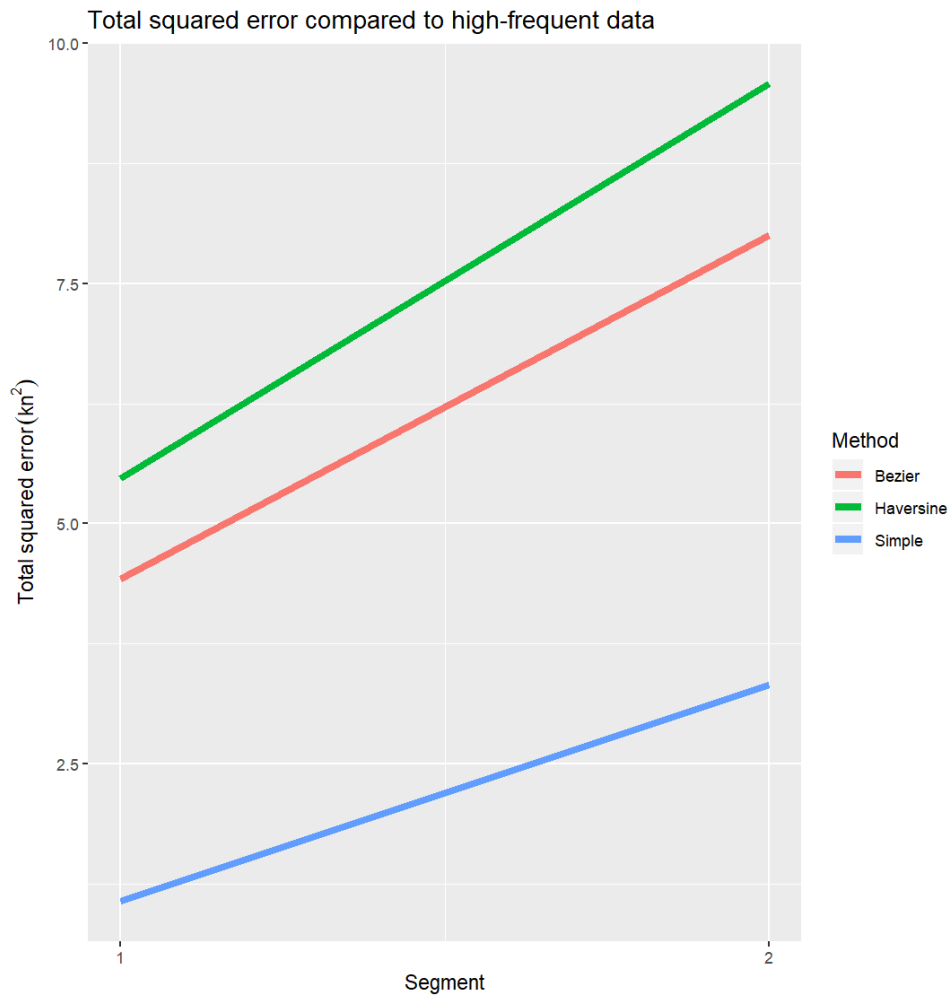


FIGURE 4.25: The total squared error of the three methods compared to the detailed data average.

This kind of behaviour however is mainly seen in the beginning period, which would explain why the simple method performs well in the beginning. The Bézier method performs well in other situations, and we therefore choose to use the Bézier method to estimate speeds.

To end this section, we show some more examples.

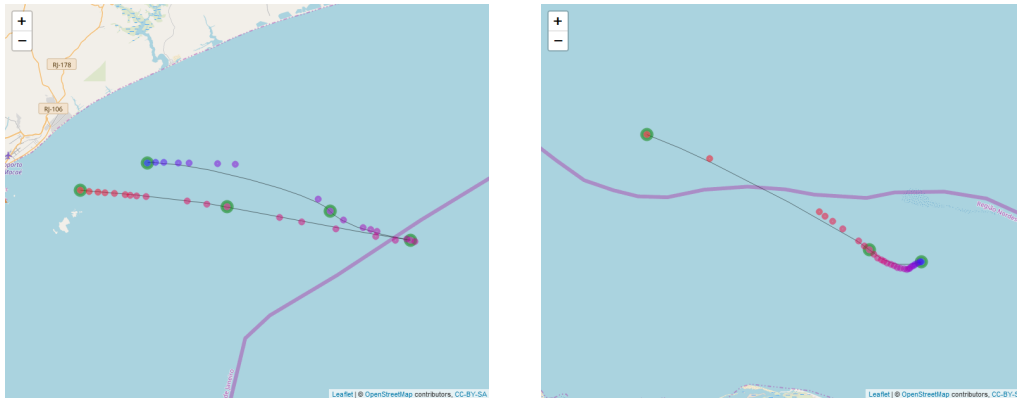


FIGURE 4.26: Two more examples of our Bézier interpolation of the non-detailed data overlaid with the detailed data. The ship is sailing with average speeds here with no high variation in sailing directions and performs reasonably well.

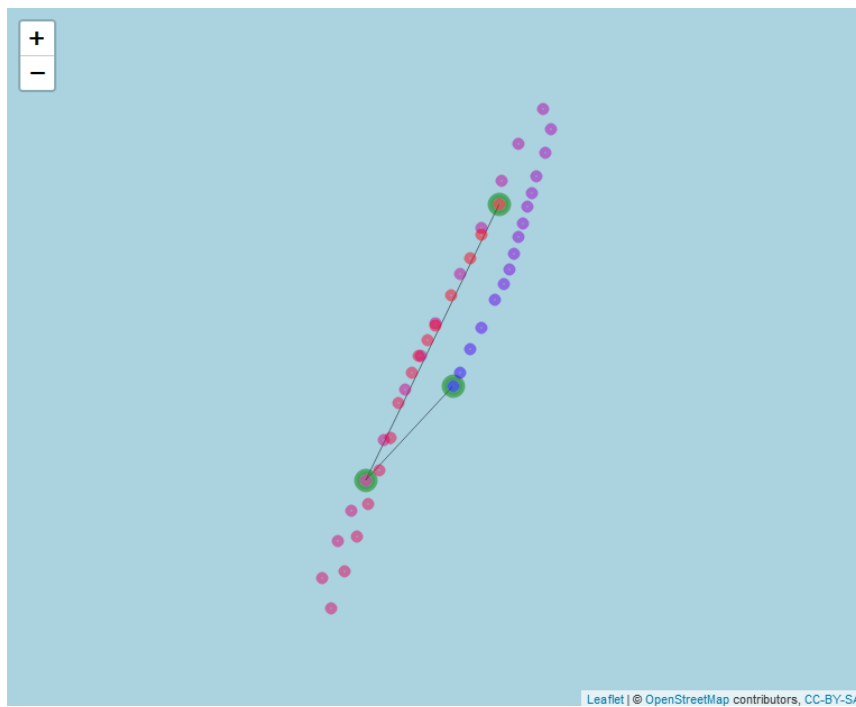


FIGURE 4.27: Another example of a case where our estimate is far from the speed estimated from the detailed set due to the zig-zagging sailing pattern of the ship at low speeds.

4.5 Final dataset

Author's note: This section is confidential

Chapter 5

Unsupervised learning

In this chapter we describe the principal component analysis and methods of clustering. All are methods of *unsupervised learning*. The difference between supervised and unsupervised problems is the existence or lack of a response variable. In supervised learning problems, a response variable is present and therefore we can for example try to predict the response based on predictor variables. In unsupervised learning problems, there is no associated response variable that we are interested in, so the aim is not to predict. Rather, the aim is the find subgroups of similar objects in the data. In this project, we are mostly interested in unsupervised learning techniques, since the data set that is available for this project has no user labels. As a matter of fact, within Damen there are no labels for customers at all. However, using clustering techniques we might be able to find groups within the data set from which we can derive useful and interpretable classes. The idea is that the variation within the groups is minimized, while simultaneously maximizing the variation between the groups. In the following sections we will make this more concrete.

5.1 Principal Component Analysis

In this section we describe a method called *Principal Component Analysis* (PCA). See James et al. (2013) and Wood (2009). PCA is a method that takes data of p (correlated) variables and finds a representation in a lower dimension whilst keeping as much of the variance as possible by converting the data into a set of uncorrelated variables called *principal components*. Often the data in \mathbb{R}^p is transformed to data in \mathbb{R}^2 , since two-dimensional data is easy to visualize. In the definition of this transformation, the first principal component has the highest variance. The second principal component is the component that has the highest variance among the components that are orthogonal on the previous component set, and so on.

More specifically, let $\tilde{\mathbf{X}} = (\tilde{X}_1, \tilde{X}_2, \dots, \tilde{X}_p)$ be a random vector of p features. Then the *first principal component* \tilde{Z}_1 is the linear combination $\tilde{Z}_1 = \phi_{11}\tilde{X}_1 + \phi_{21}\tilde{X}_2 + \dots + \phi_{p1}\tilde{X}_p$ with the largest variance. The vector $\phi_1 = (\phi_{11}, \dots, \phi_{p1})$ is the principal component loading vector and the elements in the vector are called the loadings of the first principal component and constrained to satisfy $\sum_j^p \phi_{j1}^2 = 1$. Without this constraint, arbitrarily large variances could be achieved.

Now assume we have a dataset $\mathbf{X} = (X_1, X_2, \dots, X_p)$ of n p -dimensional observations, leading to an $n \times p$ matrix. That is, \mathbf{X} is a matrix of n realizations of the p random variables in $\tilde{\mathbf{X}} = (\tilde{X}_1, \tilde{X}_2, \dots, \tilde{X}_p)$. Then let $Z_i = \phi_{11}X_1 + \phi_{21}X_2 + \dots + \phi_{p1}X_p$ be the first principal component of our data \mathbf{X} , an estimate of the real \tilde{Z}_1 . We can assume that all the variables are centered around 0 without loss of generality. For the matrix \mathbf{X} this means that all column-averages are 0. Let x_{ij} denote the realization of the j 'th

random variable of the i 'th observation. When we try to find the first principal component, it means we are trying to find a loading vector $\phi_1 = (\phi_{11}, \dots, \phi_{p1})$ so that for all n observations, we can rewrite the observation using the linear combination

$$z_{i1} = \phi_{11}x_{i1} + \phi_{21}x_{i2} + \dots + \phi_{p1}x_{ip}$$

with $i = 1, \dots, n$ in such a way that the variance on the scores z_{i1} is maximized.

Once the first principal component is found, the second principal component can then be found by finding the loading vector ϕ_2 of the second principal component. For this we look for a linear combination

$$z_{i2} = \phi_{12}x_{i1} + \phi_{22}x_{i2} + \dots + \phi_{p2}x_{ip}$$

such that the variance on the scores z_{i2} are maximized, but under the extra constraint that Z_2 is uncorrelated to Z_1 . All following principal components follow a similar structure, with the constraint that the k 'th principal component is uncorrelated to the previous $k - 1$ principal components. See appendix A for a derivation of the method.

5.2 K-Means

In this section we describe a clustering algorithm called *K-means*. See James et al. (2013) for a more rigorous explanation. The K-means algorithm is a way to divide the data into K distinct groups. Clusters achieved with this algorithm satisfy the following: Let C_1, C_2, \dots, C_K be the K clusters formed using the n observations. Then

- $C_1 \cup C_2 \cup \dots \cup C_K = \{1, \dots, n\}$
- $C_i \cap C_j = \emptyset \quad \forall i \neq j$

That is, all observations get assigned to a cluster and one cluster only. As said before, a clustering is typically considered good when the variation within the clusters is small and the variation between the clusters is large. Let C_k be cluster k and let $W(C_k)$ denote the variation of cluster C_k . Then define

$$W(C_k) = \frac{1}{|C_k|} \sum_{i \in C_k} (x_i - \bar{x}_k)^2$$

as the *within-cluster variation of cluster C_k* . In this definition, $|C_k|$ is the number of elements in C_k and $\bar{x}_k = \dots$ is the center of the cluster. The *total within-cluster variation* is then given by the sum of within-cluster variations of all clusters:

$$\text{TWSS} = \sum_k^K W(C_k)$$

Algorithm 2 Pseudocode K-means

- 1: Randomly choose K points in the \mathbb{R}^p space
 - 2: Assign every datapoint to the cluster centroid closest to the datapoint
 - 3: Calculate the new centroid for all the clusters
 - 4: Iterate steps 2 and 3 until no new assignments take place
-

The iterations of the algorithm are visualized in figure 5.1 for $p = 2$. The green dots are the initial datapoints without any clustering. The crosses are the random initialization centroids from the first step. All datapoints are assigned to the center of the cluster closest to the point. New centroids are computed and the process keeps repeating until it reaches a final clustering.

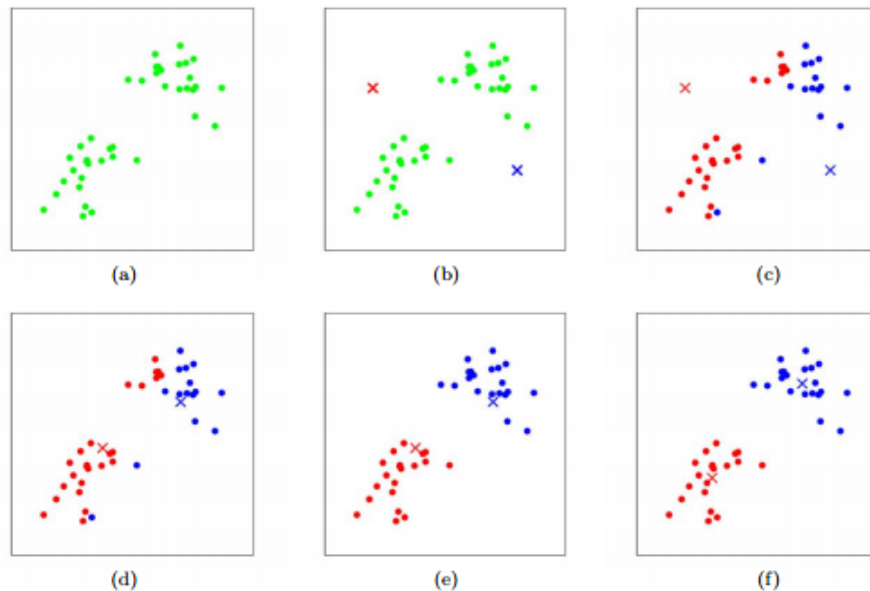


FIGURE 5.1: Steps of the K-means algorithm. Illustration taken from Pieg (2013)

Since there is only a finite number of clusterings possible and every iteration reduces the total within-cluster variation, the algorithm is guaranteed to terminate in a finite number of iterations. The final clustering however is often a local optimum. The final clustering depends on the initial randomly chosen centroids. It is therefore important to run the algorithm multiple times, and choose the clustering with the lowest TWSS. See figure 5.2.

5.2.1 Selection of K

As stated earlier, in the K-means problem the user is required to specify the number of desired clusters. However, in most problems the user is not aware of the number of clusters that might be present in the data. This also applies to our problem. We want to gain insight in the ways the ships are being used and possibly classify users. We might be able to say that we want more than 3 classifications and less than 20, but that is still far from concrete. We will explore two ways to select K.

Elbow method

We defined the total within-clusters sum of squares as

$$\text{TWSS} = \sum_k^K W(C_k) = \sum_k^K \sum_{i \in C_k} (x_i - \bar{x}_k)^2$$

This is a measure for the compactness of all the clusters and we want to minimize this value. Minimize is hard to define in this context, because surely we can just



FIGURE 5.2: Six different runs of K-means on the same data set. Different local optimums are reached and the TWSS is displayed above the image. Red numbers indicate the best TWSS (James et al., 2013)

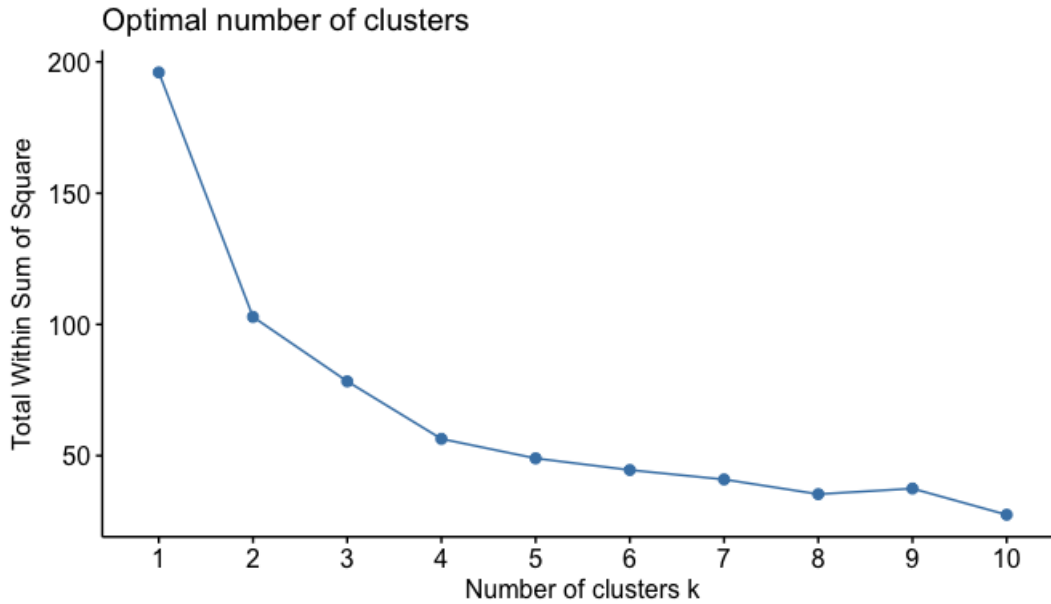


FIGURE 5.3: Plot of the elbow to determine K for a generic random dataset

take $K = n$ and make every datapoint a cluster of his own. This would lead to every cluster having $W(C_k) = 0$ and also $TWSS = 0$. However, this lead to very little practical advantages and insight. A balance must be struck between compact clusters and interpret-ability of the clusters. This is what the elbow method tries to do.

Using the elbow method you vary K over a range of values, for example 1 to 15 and calculate the TWSS for every K and plot the TWSS against the K . Typically the plot will look like figure 5.3. In this plot we can see that the nature of having more clusters leading to a lower TWSS leads to a generally decreasing line. The method got his name from the fact that we generally want to pick a K where the line shows a bend, the elbow. The idea is that for this K , the TWSS is relatively low, and choosing a higher K does not offer a significantly lower TWSS. Thus striking a balance between low TWSS and still getting interpretable results. See figure 5.3 for an example to support our explanation. The TWSS is calculated for a dataset unrelated to our research for different values of K . In this example, one could pick $K = 4$ or $K = 5$ as one can argue that is where the elbow is located.

Average silhouette method

Another method to test the compactness of the clusters and to choose K is by calculating the average silhouette value. The silhouette value is a measure of how similar an object is to object in his own cluster relative to objects in other clusters. It is defined as follows: let x_i be the i 'th object out of the dataset of n observations. Let k be the cluster x_i belongs to, $x_i \in C_k$. Then

$$a(x_i) = \frac{1}{|C_k| - 1} \sum_{j \in C_k, j \neq i} d(x_i, x_j) \quad (5.1)$$

where $d(x_i, x_j)$ can be any distance metric, although commonly the Euclidean distance is taken. In words, $a(i)$ is the average distance of x_i to all other points in the

cluster that x_i belongs to. Similarly, we define $b(i)$:

$$b(x_i) = \min_{m \neq k} \frac{1}{|C_m|} \sum_{j \in C_m} d(x_i, x_j) \quad (5.2)$$

In words, we calculate the average distance of x_i to all points in the clusters different from the cluster of x_i , and take $b(x_i)$ to be the smallest value of that. Then finally, the silhouette value is calculated as

$$s(x_i) = \begin{cases} \frac{b(x_i) - a(x_i)}{\max\{a(x_i), b(x_i)\}} & \text{if } |C_k| > 1 \\ 0 & \text{if } |C_k| = 1 \end{cases} \quad (5.3)$$

From this we can see that the silhouette value always ranges from -1 to 1 , where a value of 1 indicates a small average distance to observations in his own cluster relative to the average distance to other clusters. A negative value indicates there is another cluster with a smaller average distance to x_i than the cluster it currently belongs to. To determine the value of K , one can calculate the average silhouette value of all the datapoints and take a value of K for which the average silhouette value is maximized.

5.3 Hierarchical clustering

In this section we describe another method of clustering, called *hierarchical clustering*. Hierarchical clustering is an agglomerating method, starting from n (number of observations) clusters and ending with 1 final cluster. Any desired number of clusters can be obtained from the results in between the steps. We make this more clear in the following.

Let the objective be to cluster n observations. In hierarchical clustering, we build the clusters up. This means that in the first step, we say that we have n clusters. Then we find the two observation that are closest to each other, and merge them into one cluster, resulting in $n - 1$ clusters. For now, let the notion of "closest" be a general thought of what it means. We will define closeness of observations and clusters later once the general idea of hierarchical clustering is clear. Now again, with these $n - 1$ clusters, find the two clusters that are closest to each other and merge them again. This leads to $n - 2$ clusters. Keep repeating this, until you get one big cluster of n objects. An advantage of this method is that it is nicely presentable in what is called a *dendogram*. The method is illustrated in 5.4.

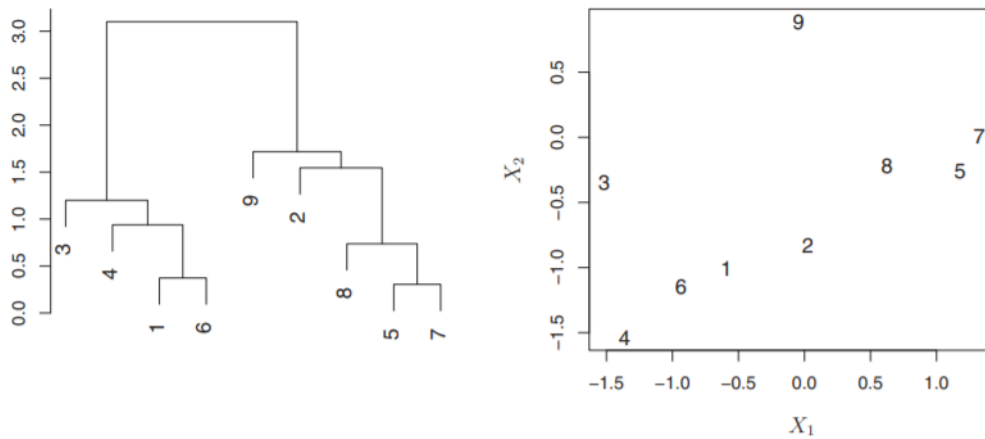


FIGURE 5.4: In this figure we show the clustering of an arbitrary two-dimensional data set to illustrate the method. On the left we see the dendrogram, and on the right the plotted data.

A dendrogram can be seen on the left in figure 5.4. We can read a dendrogram from the bottom up. In the beginning, all observations are separate clusters. Then 5 and 7 are clustered together. This is reflected in the figure on the right, as we can see that those observations lie very close to each other. After that, 1 and 6 are clustered together for the same reason. After that, 8 gets clustered with the cluster that contains 5, 7. And similarly, we work our way up until everything is grouped together. Then to form clusters, we can choose a value along the vertical axis to draw a horizontal line. Every branch stemming from the intersections would then be a cluster. So let us say that we draw a horizontal line for example at a height of 2.5. From the intersections with the dendrogram, we get two branches. That means we get two clusters: one with observations 1, 3, 4, 6, and another with the rest.

As the reader probably wondered during the steps described above, what does it mean that an observation is closest to some cluster? As mentioned earlier, we now need to define the notion of "closest". When all clusters are single observations, we can simply define this as the smallest distance between two observations given some distance metric (Euclidian for example). However, when clusters contain multiple observations there are multiple ways to define the distance.

The term for similarity between two groups of observations is called *linkage*, and for hierarchical clustering we can use different kinds of linkage. Let C_1 and C_2 be two clusters of observations, $|C_i|$ the number of elements in the clusters for $i = 1, 2$ and $d(x, y)$ some distance function on the elements of these clusters. We describe four ways to define the distance $l(C_1, C_2)$ between the two clusters.

Complete linkage

All pairwise distances of observations in cluster C_1 and C_2 are calculated. The largest distance is chosen as distance between the clusters. That is,

$$l(C_1, C_2) = \max_{p \in C_1, q \in C_2} d(p, q)$$

Single linkage All pairwise distances of observations in cluster C_1 and C_2 are calculated. The smallest distance is chosen as the distance between the clusters:

$$l(C_1, C_2) = \min_{p \in C_1, q \in C_2} d(p, q)$$

Mean linkage All pairwise distances of observations in cluster C_1 and C_2 are calculated. The average of these distances is the the distance between the clusters:

$$l(C_1, C_2) = \frac{\sum_{p \in C_1, q \in C_2} d(p, q)}{|C_1| \cdot |C_2|}$$

Centroid linkage The center of each cluster is calculated. Then the distance between the centers is taken as the cluster distance.

$$l(C_1, C_2) = d\left(\frac{\sum_{p \in C_1}}{|C_1|}, \frac{\sum_{q \in C_2}}{|C_2|}\right)$$

Most often mean linkage is used, and that is what we will be using. We will use the Euclidean distance metric on the standardized data.

5.4 Results clustering

Author's note: this section is confidential

Chapter 6

Sea state influence

In this chapter we will take a look at certain characteristics from the ship given certain conditions at sea, also called the *sea state*. Depending on applications and interests, the term sea state can refer to different statistics of the sea. In our research the term is characterized by the mean wave height per day and the mean wave period. First we will attempt to give meaning to the *availability* of a ship. After that we will take a look at the influence of the sea state on sailing speeds. Information on both subjects are of interest to Damen for various reasons. For example, the ships are being used to transport crew from land to platforms and back. If the ships are inoperable in certain sea states, operations are delayed and a significant amount of money is lost. Knowing how the *FCS 5009* performs can lead to a competitive advantage for sales if their availability is better. Or if the competitors perform better, engineers at Damen know where improvements can be made.

6.1 Availability of the ship

One performance metric of ships is their *availability*. While not exactly defined, it is interpreted as the extent to which a ship is available for sailing depending on the sea state. One way to look at this, is to examine the average sea state on a day together with whether a ship left the port that day or not. The idea behind this is that ships can opt to not leave the port on day where the current or predicted sea state is unfavourable. Note that this method is not suitable for ships that frequently travel for multiple days in a row. For now we will focus on the regions of Mexico and the Persian gulf, where ships typically make short trips that do not last longer than a day.

6.1.1 Gulf of Mexico

In this section we look at the availability of the ships in the Gulf of Mexico. We have 13 ships in our dataset that operate in this region. In figure 6.1 we put a rectangle around the area we are taking into consideration. This area contains almost all of the activity of the fleet in Mexico. For this area we obtained data on the variables *vmh0*, *vtpk*, and *vmdr*, which are the wave height, wave period, and the wave mean direction. The data spans a period from 29-02-2016 until 31-08-2018. Over this whole period, we have one data point every three hours. The data is then divided in days of 24 hours (so 8 data points) and for every day we obtain the average wave height in the highlighted area. See figure 6.2 As wave period and the wave direction are not expected to influence the availability of a ship, we will take a look at the wave height first.

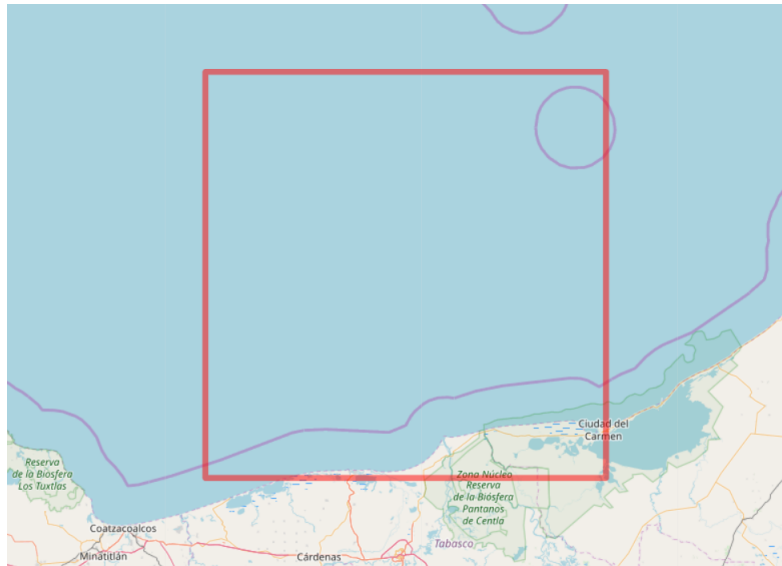


FIGURE 6.1: The area of which we get the sea state in the Gulf of Mexico

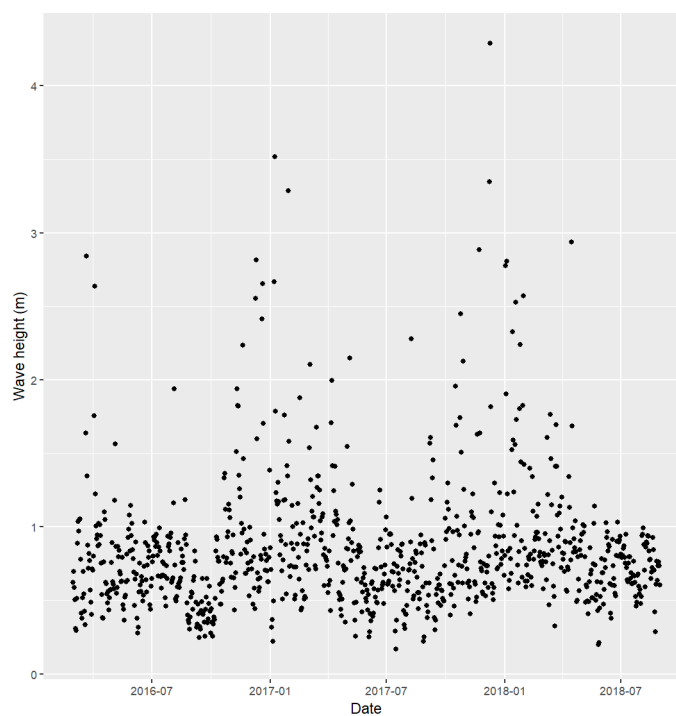


FIGURE 6.2: Average wave heights on a per day basis in the Gulf of Mexico

In the next step we mark all days on which the ship left the port. A plot showing the port leaves and the average wave height on that day can be seen in figure 6.3. In this plot, every point is a single day on which the average wave height can be seen on the x-axis. On the y-axis it shows a value close to 1 if the ship left the port that day and a value close to 0 if the ship stayed in port that day. Small perturbations have been purposefully added in the y-direction for better readability of the plot. So note that

in the figure, all y-values greater than 0.4 are actually all 1's and all y-values below 0.4 are all 0's, and those are the actual values we are working with.

Author's note: This figure is confidential

FIGURE 6.3: Port departures given wave heights for a single ship. High values on the y-axis means the ship left the port, while low value means the ship stayed in the port that day. Note that jitter is added.

We can do this for all 13 ships in Mexico, and putting them all together leads to figure 6.4. Since all the ships sail in the same region, they all have the same average wave heights on any given day. This means that for every average wave height on a day, there are multiple corresponding y-values (0 or 1) of whether the ship left the port that day or not. We average over the y-values so that every day again has one single corresponding value. The result can be seen in figure 6.5.

Author's note: This figure is confidential

FIGURE 6.4: Port departures given wave heights for all ships in Mexico. Note here again that in reality all y-values are from the set $\{0, 1\}$, but small perturbations (jitter) have been added for better readability.

Author's note: This figure is confidential

FIGURE 6.5: Port departures given wave heights for all ships in Mexico, as in figure 6.4. However in this plot the y-values of a single day are averaged out over all ships into a single value, thus representing the percentage of ships leaving the port that day.

In our application, we are interested in the probability that a ship will leave the port or will not leave the port given the wave height on any chosen day. We therefore model this as a Bernoulli process with the possible outcomes *stay* and *leave*, which stand for a ship staying in the port or the ship leaving the port respectively. We assume the distribution is dependent on the wave height h and write

$$\Pr\{\text{leave}|h\} = p(h)$$

Then Y is a Bernoulli distributed random variable and represents the availability of a ship on a day with wave height h ,

$$Y = \begin{cases} 1 & \text{with probability } p(h) \text{ leave port} \\ 0 & \text{with probability } 1 - p(h) \text{ stay in port} \end{cases}$$

If we denote our data with Y_i the availability of a ship on a certain day and h_i the wave height on that day, then we have n independent observations y_i for $i = 1, \dots, n$ where $Y_i \sim \text{Bern}(p(h_i))$. We are interested in estimating $p(h)$ using the data that we have.

The idea behind availability is that it decreases as waves get higher. In this case, it would mean that ships tend to leave the port less often as the waves get higher. We therefore perform monotonic regression on the data of figure 6.5 to obtain $p(h)$.

6.1.2 Monotonic regression

Monotonic regression is a technique of fitting a line to observations where similar to other regression techniques, the line should fit the observations as closely as possible. In monotonic regression there is however also the extra constraint that the line should be non-decreasing or non-increasing everywhere, also called isotonic or anti-tonic respectively. We make this idea more explicit in the following.

Let $(x_i, y_i) \in \mathbb{R}^2$ be our n observations for $i = 1, 2, \dots, n$ and $\mathbf{w} \in \mathbb{R}^n$ a weight vector. Then $\mathbf{z} = (z_1, \dots, z_n)$ is the vector that minimizes

$$\sum_i^n w_i (y_i - z_i)^2 \tag{6.1}$$

where z_1, \dots, z_n are the maximum likelihood estimators under the inequality constraints $z_1 \geq z_2 \geq \dots \geq z_n$ for an anti-tonic fit See appendix G for a further discussion of a monotonic fit.

In our case, the y_i is the percentage of ships that left the port on a certain day and the w_i is the number of ships of which we have data on that day.

In figure 6.6 we can see the result of the isotonic fit on the ships in Mexico.

Author's note: This figure is confidential

FIGURE 6.6: An antitonic fit in red on the data as displayed in figure 6.5

In the following, the result of the monotonic fit, which is our estimate of the probability that a ship will leave the port given the wave height, is subjected to a test. The red line shows a fairly rapid decent from wave heights higher than 1.5 meters, and we would like to test whether this result is significant or likely just a result of chance. For this we establish the following null-hypothesis:

H_0 : *The availability of a ship is not dependent on the wave height, but constant.*

and the alternative hypothesis

H_1 : *The availability of a ship decreases as the wave height increases.*

For the test, we take 2000 permutations of the original y-values in figure 6.4 without the perturbations. We then again take the average y-value of all points that share the same x-value and perform an isotonic regression.

Let n be the number of data points that we have. Then x and y are both vectors of size n , where x is the vector of average wave heights per day, and y is the ratio of ships leaving the port on a day. After performing an isotonic regression, we get a vector \hat{p} of size n where \hat{p}_i is the predicted percentage of ships leaving given the wave height x_i , where $i = 1, \dots, n$. If \bar{y} is the mean of y , then we can compute the test statistic

$$T = \sum_i^n (p_i - \bar{y})^2$$

If in reality the ships stay in port more often during heavy weather, we expect T to be big. If there is no relation between setting sail and the weather, then the predicted value will stray not too far from the average value and we expect T to be small. This is illustrated in figure 6.7, where we show a plot similar to figure 6.6 but calculated for one of the permutations. From this figure we can calculate T , which gives us the statistic on this one permutation. In this case we have $n = 915$ which gives 915! possible permutations. We obtain 2000 random permutations out of all possibilities.

Author's note: This figure is confidential

FIGURE 6.7: Isotonic regression on a permutation of the original port leaves of all ships. The red line is $\hat{\rho}$ and the black line is \bar{y}

6.1.3 Persian Gulf

Author's note: This section is confidential

6.1.4 The Caribbean

Author's note: This section is confidential

6.1.5 Nigeria

Author's note: This section is confidential

6.1.6 Conclusion availability of ships

Author's note: This section is confidential

6.2 Speed on established routes

There are many variables that influence the speed of a ship, most of which we do not have access too. These variables might be the weight of the load, captain on the ship, sailing through protected areas and many more. In an effort to maintain a base for comparison sake, we isolate an established path and analyse the speeds on that path.

We will take a look at the path shown in figure 6.8.

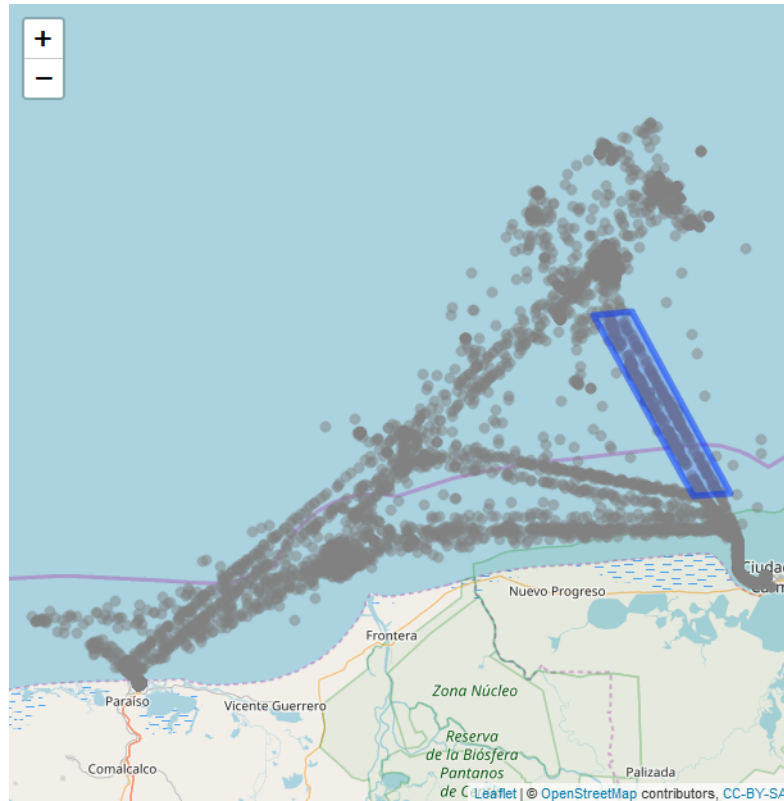


FIGURE 6.8: One path in Mexico shown for ship 2

From the 13 ships in our fleet that operate in the Mexican Gulf, 10 ships have significant activity on the selected path. From those 10 ships together, we have 1983 data points in on this path. Do note that for this section, we are looking at the point speeds as reported by the AIS system, and not the average speeds on hourly sections calculated using a Bézier curve as used in earlier sections. The reason for this is that when trying to cluster ships, we are trying to find general behaviour that we can use to characterize a ship, which includes the speed in between measurement points. In this section however, we are interested in the relationship between the speed and the sea state, so we feel the precise speed at a single point is more suitable for this analysis.

Author's note: This section is confidential

In multiple regression, we try to find a linear relation between a dependent variable and multiple independent variables. The regression model is as follows:

$$s_i = \beta_0 + \beta_1 h_i + \beta_2 \theta_i + \beta_3 \phi_i + \epsilon_i$$

where for $i = 1, \dots, n$ (for $n = 1983$, the number of data points we have on this path)

- s_i is the speed of point i
- h_i is the wave height at the location of the ship
- θ_i is the angle between the direction of movement from the ship and the wave direction

- ϕ_i is the period of the waves
- $\beta_{0,1,2,3}$ are the regression parameters
- ϵ_i is a random error-term with mean zero

Using the $lm()$ function in the R software we perform a multiple regression with the model as described above.

Author's note: This section is confidential

Chapter 7

Conclusions

Now that we have done our analyses, we come back to our goal of the study and the supporting questions we set up for ourselves at the beginning. DAMEN wants to know how the ships they have sold are being used. Understanding their usage can lead to both ideas for new products, but also shortcomings in the current products. So our goal was stated as: *Use statistical techniques on the data to gain useful and interpretable insights in the behaviour of the customers.* To achieve this goal, we set out to answer these two questions:

- How can we divide our fleet of ships in groups with similar behaviour within the group?
- Are characteristics of ship use influenced by conditions at sea?

We first take a look at the first question.

After enriching our data, the K-Means clustering algorithm and a bootstrap method with the Jaccard index as statistic lead to 6 stable clusters. These clusters very similar to clusters found using a hierarchical clustering method.

These six clusters can be seen as classes that we can assign to current and possibly future ships. Generally, these six clusters are combinations of how often the ships are being used, and the duration of most trips for a ship.

Furthermore, to get a more detailed description of the ship operations in arbitrary periods of time, we can classify the trips of a single ship instead of trying to classify the ships themselves. Using the same methods as we did on the ship statistics, we get trip clusters. This gives a more natural way of classes of ships, that provides more insights in the opinion of the author.

For the second question, we had a look at both the availability of the ships, and the sailing speeds. *Author's note: This section is confidential*

Chapter 8

Recommendations

This is one of the first data-driven research that is being done at Damen, with data that has not been analyzed before. Being the first research of this kind, the intention was more exploratory than advisory. Results from this research is not primarily meant to lead to big changes in the company, but rather to give people at Damen ideas for research, understanding of certain activities, and direction of thoughts.

First, we can say that a higher resolution would lead to better results. Furthermore, there are more questions we can answer with a higher resolution. Do ships sail faster when they go to a platform further away? How long do ships stay at platforms? Is it altered by the sea state? Depending on the question you try to answer, different resolutions are minimally required. While a resolution of one data point per hour is lacking, one per minute would be too detailed for most questions. We believe a resolution of 1 per 10 minutes would already improve the analysis a lot.

Second, our research shows that by dividing the data into blocks of data that all represent a single trip, we can make clusters of distinct ship behaviour based on those trip statistics. However, we believe that more data would lead to better clusters. We can for example purchase a NAVIONICS map, and combine it with the AIS data. That way, most of the locations can be labeled automatically. See for example figures 8.1 and 8.2. In these figures we can see areas that are labeled as anchorage areas and wind parks. We are not sure how a paid map works, but if one can enter coordinates and receive whether those coordinates are situated in a labeled area, then combining these two data sets can be very straightforward.

Once that is done, we can for example divide all time over the different locations and have variables showing what percentage of the time they spend at each location in a certain period. Then we can use any of the methods we explored in this report to show and cluster the activities of the ship based on the locations labels.

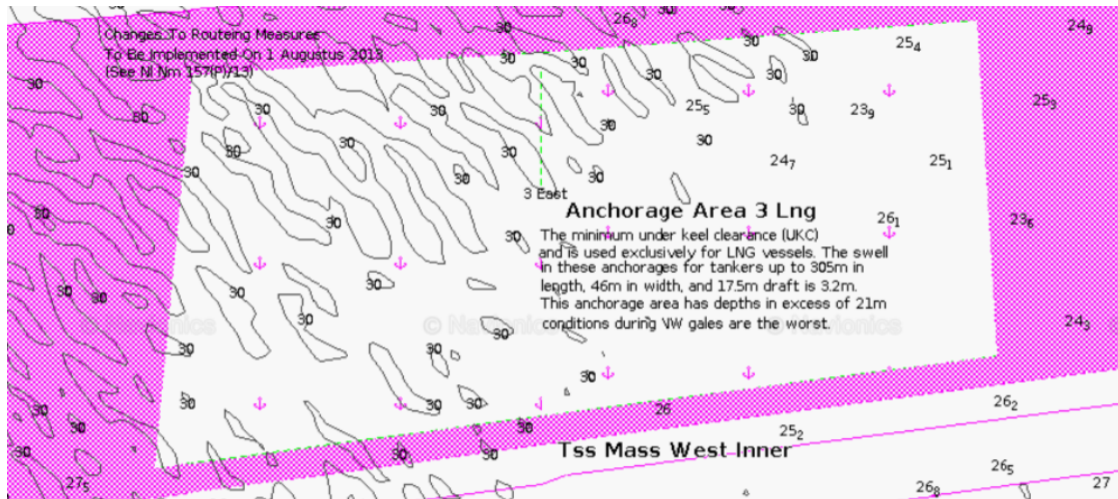


FIGURE 8.1: Area on the NAVIONICS map that shows all anchorage areas, which are mooring positions.

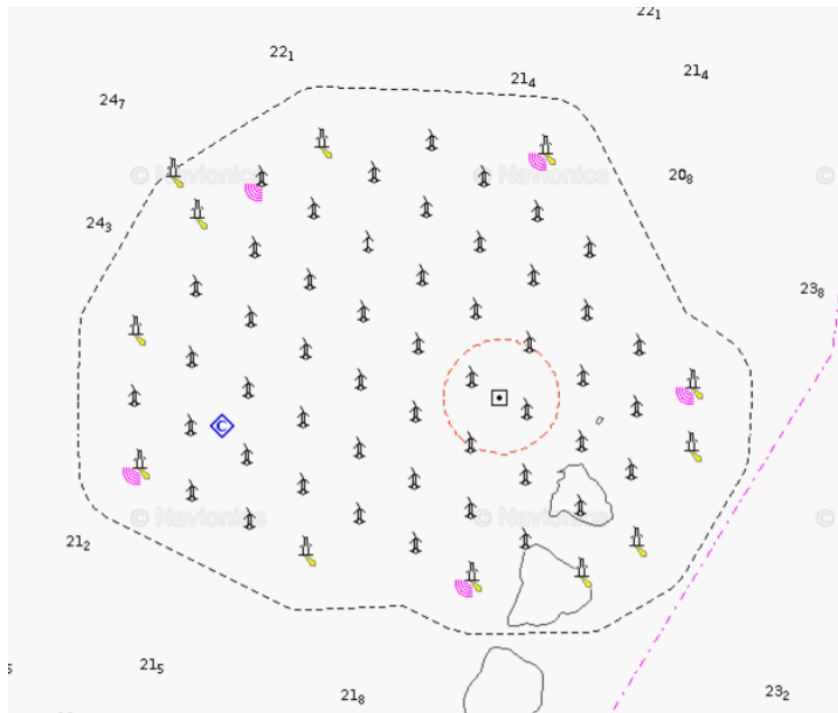


FIGURE 8.2: Area on the NAVIONICS map showing wind turbines on a wind farm.

In the end, unsupervised problem results are not always easy to interpret. Clusters can be useful or useless, and depend on what one is trying to achieve. With this research we have shown that using clustering algorithms on the variables that we defined lead to stable clusters, but might represent useful clusters because of the many mooring points.

We believe this thesis gives a better idea of what is possible, and what kind of results one might find using the clustering algorithms. However, the clusters one finds depend very much on the variables used. We hope that after reading this thesis, researchers at DAMEN have a good idea of the possibilities of clustering, and that it

will lead to insightful discussions about what the context really is, and what kind of clusters or patterns they want to find. By answering these questions, it will become much clearer what kind of data is needed (e.g. higher-frequency, labeled, engine data), and in extension what is needed to acquire that data. Then using the techniques that we explored in this thesis, the results will likely be more useful.

Appendix A

All data variables

Author's note: This section is confidential

TABLE A.1: All provided variables and their explanation

Variable	Explanation	Unit
TIME	AIS time	[UTC time]
LAT	AIS latitude	[deg]
LON	AIS longitude	[deg]
SPEED	AIS speed	[kn]
COURSE	AIS course	[deg]
HEADING	AIS heading	[deg]
STATUS	AIS vessel status	[-]
IMO	AIS IMO number	[-]
FRACTION	Fraction of time available in AIS	[-]
wav_TIME	WAVE data time	[UTC time]
wav_LAT	WAVE data latitude	[deg]
wav_LON	WAVE data longitude	[deg]
VHM0	Significant wave height	[m]
VMDR	Wave mean direction	[deg]
VTPK	Peak period wave spectrum	[s]
win_TIME	WIND data time	[UTC time]
win_LAT	WIND data latitude	[deg]
win_LON	WIND data longitude	[deg]
eastward_wind	eastward wind speed	[m/s]
northward_wind	northward wind speed	[m/s]
eastward_wind_rms	eastward wind speed RMS	[m/s]
northward_wind_rms	northward wind speed RMS	[m/s]
phy_TIME	PHYSICS data time	[UTC time]

TABLE A.1: All provided variables and their explanation

Variable	Explanation	Unit
phy_LAT	PHYSICS data latitude	[deg]
phy_LON	PHYSICS data longitude	[deg]
thetao	water temperature surface	[C]
so	salinity surface	[psu]
zos	sea surface height above geo ID	[m]
uo	eastward current velocity	[m/s]
vo	northward current velocity	[m/s]
siconc	sea ice concentration	[-]
sithick	sea ice thickness	[m]
usi	eastward sea ice velocity	[m/s]
vsi	northward sea ice velocity	[m/s]
bio_TIME	BIOCHEMICAL data time	[UTC time]
bio_LAT	BIOCHEMICAL data latitude	[deg]
bio_LON	BIOCHEMICAL data longitude	[deg]
O2	mole concentration of dissolved oxygen	[mmol/m3]
DATA_FRACTION_DOWNLOADED	fraction of downloaded data per environmental product	[-]

Appendix B

Raw data analysis

Author's note: This section is confidential

Appendix C

Sea state plots without ports

Author's note: This section is confidential

Appendix D

Bézier curves

We prove the endpoint interpolation property and the endpoint tangent property here. **Proof endpoint interpolation property**

Let $\mathbf{b}_0, \mathbf{b}_1, \dots, \mathbf{b}_n \in \mathbb{R}^2$ be the control points for the Bézier curve $\mathbf{B}(t)$ of degree n . Then we can write

$$\mathbf{B}(t) = \sum_{i=0}^n \mathbf{b}_i B_{i,n}(t) \quad \text{for } t \in [0, 1]$$

where

$$B_{i,n}(t) = \begin{cases} \frac{n!}{(n-i)!i!} (1-t)^{n-i} t^i & \text{if } 0 \leq i \leq n \\ 0 & \text{otherwise} \end{cases}$$

Note that by definition we have that $0! = 1$ and $0^0 = 1$. Evaluating the Bernstein Polynomials for $t = 0$ we get for $i = 0$ and $n > 0$:

$$\begin{aligned} B_{0,n}(0) &= \frac{n!}{(n-0)!0!} (1-0)^{n-0} 0^0 \\ &= \frac{n!}{n!} (1-0)^n \\ &= 1 \end{aligned}$$

For the other polynomials with $0 < i \leq n$, $n > 0$ we get:

$$\begin{aligned} B_{i,n}(0) &= \frac{n!}{(n-i)!i!} (1-0)^{n-0} 0^i \\ &= \frac{n!}{(n-i)!i!} 0^i \\ &= 0 \end{aligned}$$

Then we get for $\mathbf{B}(0)$:

$$\begin{aligned} \mathbf{B}(0) &= \sum_{i=0}^n \mathbf{b}_i B_{i,n}(0) \\ &= \mathbf{b}_0 B_{0,n}(0) \\ &= \mathbf{b}_0 \end{aligned}$$

For $t = 1$ we equally find that for $0 \leq i < n, n > 0$

$$\begin{aligned} B_{i,n}(1) &= \frac{n!}{(n-i)!i!} (1-1)^{n-i} 1^i \\ &= \frac{n!}{(n-i)!i!} 0^{n-i} \\ &= 0 \end{aligned}$$

and for $i = n, n > 0$

$$\begin{aligned} B_{n,n}(1) &= \frac{n!}{(n-n)!n!} (1-1)^{n-n} 1^n \\ &= \frac{n!}{n!} 0^0 \\ &= 1 \end{aligned}$$

leading to $B(1)$:

$$\begin{aligned} B(1) &= \sum_{i=0}^n b_i B_{i,n}(1) \\ &= b_n B_{n,n}(1) \\ &= b_n \end{aligned}$$

□

Proof Endpoint Tangent Property

We try to find the first derivative of $B(t)$, which we recall is defined as

$$B(t) = \sum_{i=0}^n b_i B_{i,n}(t)$$

. Since the control points are constant and do not depend on t , we are hence concerned with finding the derivatives of the Bernstein polynomials $B_{i,n}(t)$, defined by

$$B_{i,n}(t) = \frac{n!}{(n-i)!i!} (1-t)^{n-i} t^i$$

Taking the derivative w.r.t t yields:

$$\begin{aligned} \frac{d}{dt} B_{i,n}(t) &= \frac{d}{dt} \frac{n!}{(n-i)!i!} (1-t)^{n-i} t^i \\ &= -(n-i) \frac{n!}{(n-i)!i!} (1-t)^{n-i-1} t^i + i \frac{n!}{(n-i)!i!} (1-t)^{n-i} t^{i-1} \\ &= -\frac{n!}{(n-i-1)!i!} (1-t)^{n-i-1} t^i + \frac{n!}{(n-i)!(i-1)!} (1-t)^{n-i} t^{i-1} \quad (\text{D.1}) \\ &= n \frac{(n-1)!}{(n-i-1)!i!} (1-t)^{n-i-1} t^i + n \frac{(n-1)!}{(n-i)!(i-1)!} (1-t)^{n-i} t^{i-1} \\ &= -n B_{i,n-1}(t) + n B_{i-1,n-1}(t) \\ &= n (B_{i-1,n-1}(t) - B_{i,n-1}(t)) \end{aligned}$$

Now we state that the following:

$$\mathbf{B}'(t) = \sum_{i=1}^{n-1} \mathbf{b}_i^{(1)} B_{i,n-1}(t) \quad (\text{D.2})$$

where $\mathbf{b}_i^{(1)} = n(\mathbf{b}_{i+1} - \mathbf{b}_i)$.

Proof

From equation D.1 we know that $B'_{i,n}(t) = n(B_{i-1,n-1}(t) - B_{i,n-1}(t))$. Then using the fact that $B_{-1,n-1}(t) = B_{n,n-1} = 0$ (as defined in 4.2) we get:

$$\begin{aligned} \mathbf{B}'(t) &= \sum_{i=0}^n \mathbf{b}_i B'_{i,n}(t) \\ &= \sum_{i=0}^n \mathbf{b}_i n (B_{i-1,n-1}(t) - B_{i,n-1}(t)) \\ &= \sum_{i=0}^n n \mathbf{b}_i B_{i-1,n-1}(t) - \sum_{i=0}^n n \mathbf{b}_i B_{i,n-1}(t) \\ &= \sum_{i=1}^n n \mathbf{b}_i B_{i-1,n-1}(t) - \sum_{i=0}^{n-1} n \mathbf{b}_i B_{i,n-1}(t) \\ &= \sum_{i=0}^{n-1} n \mathbf{b}_{i+1} B_{i,n-1}(t) - \sum_{i=0}^{n-1} n \mathbf{b}_i B_{i,n-1}(t) \\ &= \sum_{i=0}^{n-1} n(\mathbf{b}_{i+1} - \mathbf{b}_i) B_{i,n-1}(t) \end{aligned}$$

From the last result it follows that $\mathbf{B}'(0) = n(\mathbf{b}_1 - \mathbf{b}_0)$ and $\mathbf{B}'(1) = n(\mathbf{b}_n - \mathbf{b}_{n-1})$.

□

Appendix E

Principal Component Analysis

In this appendix we describe the procedure to find the principal components and a derivation of our steps.

Let $\tilde{\mathbf{X}} = (\tilde{X}_1, \tilde{X}_2, \dots, \tilde{X}_p)$ be a random vector of p features. Let $\mathbf{X} = (\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_p)$ be a dataset of n observations on the p variables, leading to a $n \times p$ matrix. We can not find the principal component loadings of $\tilde{\mathbf{X}}$, but instead estimate the loadings using sample \mathbf{X} . Let x_{ij} denote the realization of the j 'th random variable of the i 'th observation and let $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{ip})$. When we try to find the first principal component, it means we are trying to find a loading vector $\boldsymbol{\phi}_1 = (\phi_{11}, \dots, \phi_{p1})$ so that for all n observations, we can rewrite the observation using the linear combination

$$z_{i1} = \phi_{11}x_{i1} + \phi_{21}x_{i2} + \dots + \phi_{p1}x_{ip}$$

with $i = 1, \dots, n$ in such a way that the variance on the scores z_{i1} is maximized, under the constraint $\sum_j^p \phi_{j1}^2 = 1$.

Let $\mathbf{z}_1 = (z_{11}, z_{21}, \dots, z_{n1})$, ie the scores of the data on the first principal component. We then look for for a loadings vector that maximizes $\text{Var}(\mathbf{z}_1)$:

$$\text{Var}(\mathbf{z}_1) = \frac{1}{n} \sum_{i=1}^n (z_{i1} - \bar{z}_1)^2 \quad (\text{E.1})$$

Writing out the mean \bar{z}_1 we get:

$$\begin{aligned} \bar{z}_1 &= \frac{1}{n} \sum_{i=1}^n z_{i1} \\ &= \frac{1}{n} \sum_{i=1}^n \sum_{j=1}^p \phi_{j1} x_{ij} \\ &= \frac{1}{n} \sum_{j=1}^p \sum_{i=1}^n \phi_{j1} x_{ij} \end{aligned} \quad (\text{E.2})$$

Without loss of generality, assume that the data has been normalized such that the mean of every variable is zero. Then for every $j = 1, \dots, p$ we have that $\sum_{i=1}^n \phi_{j1} x_{ij} = 0$

which implies that $\bar{\mathbf{z}}_1 = 0$. Using the result from E.2 we can rewrite E.1 as

$$\begin{aligned}
\text{Var}(\mathbf{z}_1) &= \frac{1}{n} \sum_{i=1}^n (z_{i1})^2 \\
&= \frac{1}{n} \sum_{i=1}^n \sum_{j=1}^p (\phi_{j1} x_{ij})^2 \\
&= \frac{1}{n} \sum_{i=1}^n (\boldsymbol{\phi}_1 \cdot \mathbf{x}_i)^2 \\
&= \frac{1}{n} (\mathbf{X} \boldsymbol{\phi}_1)^T (\mathbf{X} \boldsymbol{\phi}_1) \\
&= \frac{1}{n} \boldsymbol{\phi}_1^T \mathbf{X}^T \mathbf{X} \boldsymbol{\phi}_1 \\
&= \boldsymbol{\phi}_1^T \frac{\mathbf{X}^T \mathbf{X}}{n} \boldsymbol{\phi}_1 \\
&= \boldsymbol{\phi}_1^T \mathbf{V} \boldsymbol{\phi}_1
\end{aligned} \tag{E.3}$$

where in the last line we rewrote $\frac{\mathbf{X}^T \mathbf{X}}{n}$ as \mathbf{V} . The problem can thus be redefined as finding $\boldsymbol{\phi}_1$ that maximizes $\boldsymbol{\phi}_1^T \mathbf{V} \boldsymbol{\phi}_1$ under the constraint $\sum_j^p \phi_{j1}^2 = \boldsymbol{\phi}_1^T \boldsymbol{\phi}_1 = 1$.

We can solve this problem using the Lagrange multiplier λ if we rewrite the constraint as $\boldsymbol{\phi}_1^T \boldsymbol{\phi}_1 - 1 = 0$. Using the method of Lagrange we get

$$\mathcal{L} = \boldsymbol{\phi}_1^T \mathbf{V} \boldsymbol{\phi}_1 - \lambda (\boldsymbol{\phi}_1^T \boldsymbol{\phi}_1 - 1) \tag{E.4}$$

and

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\phi}_1} = 2\mathbf{V} \boldsymbol{\phi}_1 - 2\lambda \boldsymbol{\phi}_1 \tag{E.5}$$

Setting the derivative to zero yields:

$$\begin{aligned}
2\mathbf{V} \boldsymbol{\phi}_1 - 2\lambda \boldsymbol{\phi}_1 &= 0 \\
\implies 2\mathbf{V} \boldsymbol{\phi}_1 &= 2\lambda \boldsymbol{\phi}_1 \\
\implies \mathbf{V} \boldsymbol{\phi}_1 &= \lambda \boldsymbol{\phi}_1
\end{aligned} \tag{E.6}$$

From this we can see that $\boldsymbol{\phi}_1$ is an eigenvector of the matrix \mathbf{V} . And since

$$\begin{aligned}
\mathbf{V} \boldsymbol{\phi}_1 &= \lambda \boldsymbol{\phi}_1 \\
\implies \boldsymbol{\phi}_1^T \mathbf{V} \boldsymbol{\phi}_1 &= \boldsymbol{\phi}_1^T \lambda \boldsymbol{\phi}_1 \\
\implies \boldsymbol{\phi}_1^T \mathbf{V} \boldsymbol{\phi}_1 &= \lambda
\end{aligned} \tag{E.7}$$

we can see that the loading vector of the first principal component $\boldsymbol{\phi}_1$ is the eigenvector associated with the maximum eigenvalue.

To find the second principal component, we again try to maximize

$$\text{Var}(\mathbf{z}_2) = \boldsymbol{\phi}_2^T \mathbf{V} \boldsymbol{\phi}_2 \tag{E.8}$$

but this time under the extra constraint that $\text{Cov}(z_1, z_2) = 0$ besides the constraint $\phi_2^T \phi_2 = 1$. Similar to equation E.3 we get that

$$\begin{aligned}
\text{Cov}(z_1, z_2) &= \frac{1}{n} \sum_{i=1}^n (\phi_1 x_i) (\phi_2 x_i) \\
&= \frac{1}{n} (X \phi_1)^T (X \phi_2) \\
&= \phi_1^T V \phi_2 \\
&= \phi_2^T V \phi_1 \\
&= \phi_2^T \lambda \phi_1 \quad \text{Substitute E.6} \\
&= \lambda \phi_2^T \phi_1 \\
&= \lambda \phi_1^T \phi_2
\end{aligned} \tag{E.9}$$

So we can rewrite our constraint as $\lambda \phi_1^T \phi_2 - 1 = 0$ set up the Lagrangian with the two constraint as:

$$\mathcal{L} = \phi_2^T V \phi_2 - \lambda_2 (\phi_2^T \phi_2 - 1) - \theta (\phi_1^T \phi_2) \tag{E.10}$$

where we rewrite the product of the first Lagrangian multiplier λ and the new multiplier as θ . Like in E.5 we take the derivative and set it equal to zero:

$$\begin{aligned}
\frac{\partial \mathcal{L}}{\partial \phi_2} &= 2V \phi_2 - 2\lambda_2 \phi_2 - \theta \phi_1 = 0 \\
&\implies 2\phi_1^T V \phi_2 - 2\lambda_2 \phi_1^T \phi_2 - \theta \phi_1^T \phi_1 = 0 \\
&\implies 2\phi_1^T V \phi_2 - 2\lambda_2 \phi_1^T \phi_2 - \theta = 0 \\
&\implies 0 - 0 - \theta = 0 \\
&\implies \theta = 0
\end{aligned} \tag{E.11}$$

Note that in the third and fourth line we use the constraint and the result from E.9. The Lagrangian equation then becomes

$$\begin{aligned}
2V \phi_2 - 2\lambda_2 \phi_2 &= 0 \\
\implies V \phi_2 &= \lambda_2 \phi_2
\end{aligned} \tag{E.12}$$

which shows again that ϕ_2 is an eigenvector of V . However, we cannot choose the eigenvector associated with the biggest eigenvalue, since we already did that for ϕ_1 . So for ϕ_2 we choose the eigenvector associated with the second biggest eigenvalue. Following principal components can be found in a similar fashion.

Appendix F

Results bootstrap of PCA on the ships

Author's note: This section is confidential

Appendix G

Isotonic regression

In this chapter we describe the methods for an isotonic regression. See Groeneboom and Jongbloed (2014) for reference. Let $y \in \mathbb{R}^n$, and C_n the closed convex cone in \mathbb{R}^n defined by

$$C = \{x = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n \mid x_1 \leq x_2 \leq \dots \leq x_n\}$$

If $w = (w_1, w_2, \dots, w_n) \in (0, \infty)^n$ is a weight vector, then in an isotonic regression the solution r is

$$r = \arg \min_{x \in C_n} \sum_i^n (y_i - x_i)^2 w_i \quad (\text{G.1})$$

If we define $Q(x)$ as

$$Q(x) = \frac{1}{2} \sum_i^n (y_i - x_i)^2 w_i$$

then we can rewrite the solution r as

$$r = \arg \min_{x \in C_n} Q(x) \quad (\text{G.2})$$

We state that for the solution r , the following must hold:

$$\begin{aligned} r \text{ minimizes } Q(x) \text{ over the convex cone } C_n \\ \iff \\ \sum_{j=1}^i r_j w_j = \begin{cases} = \sum_{j=1}^i y_j w_j & \forall i = 1, 2, \dots, n \\ \leq \sum_{j=1}^i y_j w_j & \text{if } r_{i+1} > r_i \text{ or } i = n \end{cases} \end{aligned} \quad (\text{G.3})$$

Proof

First, assume that r minimizes $Q(x)$ over the convex cone C_n . Note that $Q(x)$ is a strictly convex function on \mathbb{R}^n , so it has a unique unique minimizer on C_n . Define for $i = 1, 2, \dots, n$ the vectors $v^{(i)}$ by $v_j^{(i)} = 1_{\{1, 2, \dots, i\}}(j)$. To illustrate, if $i = 3$, then we have that $v^{(i)} = (1, 1, 1, 0, 0, \dots, 0) \in \mathbb{R}^n$. Then we have that $r - \epsilon v^{(i)} \in \mathbb{R}^n$ for all

$\epsilon > 0$. Since $Q(x)$ is convex and r is the minimizer over C_n , we get for every i :

$$\begin{aligned}
0 &\leq \lim_{\epsilon \downarrow 0} \frac{Q(r - \epsilon v^{(i)}) - Q(r)}{\epsilon} \\
&= \lim_{\epsilon \downarrow 0} \frac{Q(r - \epsilon(1, \dots, 1, 0, \dots, 0)) - Q(r)}{\epsilon} \\
&= \frac{1}{2} \lim_{\epsilon \downarrow 0} \frac{\sum_{j=1}^i (r_j - \epsilon - y_j)^2 w_j + \sum_{j=i+1}^n (r_j - y_j)^2 w_j - \sum_{j=1}^n (r_j - y_j)^2 w_j}{\epsilon} \\
&= \frac{1}{2} \lim_{\epsilon \downarrow 0} \frac{\sum_{j=1}^i (r_j - \epsilon - y_j)^2 w_j - \sum_{j=1}^i (r_j - y_j)^2 w_j}{\epsilon} \tag{G.4} \\
&= \frac{1}{2} \lim_{\epsilon \downarrow 0} \frac{\sum_{j=1}^i ((r_j - \epsilon - y_j)^2 w_j - (r_j - y_j)^2 w_j)}{\epsilon} \\
&= \frac{1}{2} \lim_{\epsilon \downarrow 0} \frac{\sum_{j=1}^i -2\epsilon(r_j - y_j)w_j + \epsilon^2 w_j}{\epsilon} \\
&= \sum_{j=1}^i (y_j - r_j)w_j
\end{aligned}$$

For all i such that $r_{i+1} > r_i$ or $i = n$, it must also hold that $r + \epsilon v^{(i)} \in \mathbb{R}^n$ for $\epsilon > 0$ sufficiently small enough. Then we have for all such i that

$$\begin{aligned}
0 &\leq \lim_{\epsilon \downarrow 0} \frac{Q(r + \epsilon v^{(i)}) - Q(r)}{\epsilon} \\
&= \sum_{j=1}^i (y_j - r_j)w_j \tag{G.5}
\end{aligned}$$

where all steps are similar as in G.4. Then taken together with the inequality in the other direction, we have proven the \Rightarrow part of the statement.

For the \Leftarrow part, we note that an r that satisfies these and (in)equalities can be constructed. We are looking for a vector r that satisfies

$$\sum_{j=1}^i r_j w_j = \begin{cases} = \sum_{j=1}^i y_j w_j & \forall i = 1, 2, \dots, n \\ \leq \sum_{j=1}^i y_j w_j & \text{if } r_{i+1} > r_i \text{ or } i = n \end{cases}$$

Define the cumulative sum diagram $P_0 = (0, 0)$ and $P_i = \left(\sum_{j=1}^i w_j, \sum_{j=1}^i y_j w_j \right) \in \mathbb{R}^2$ for $i = 1, \dots, n$. Then create the greatest convex minorant of these points. Then r_i is given by the left derivative of this convex minorant evaluated at point P_i . Then by this construction, we see that such an r exists and is unique, thereby proving the \Leftarrow part.

Bibliography

- Alevizos, Elias, Alexander Artikis, and Georgios Paliouras (2017). “Event Forecasting with Pattern Markov Chains”. In: *DEBS*.
- Bijman, Jeroen (2017). “Cluster driving behaviour and assigning clusters to safe and unsafe driving behaviour through raw GPS trajectory data.” Master’s Thesis. Tilburg University.
- Bonham, Christopher et al. (2018). “Analysing port and shipping operations using big data”. In:
- Ester, Martin et al. (1996). “A Density-based Algorithm for Discovering Clusters a Density-based Algorithm for Discovering Clusters in Large Spatial Databases with Noise”. In: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*. KDD’96. Portland, Oregon: AAAI Press, pp. 226–231. URL: <http://dl.acm.org/citation.cfm?id=3001460.3001507>.
- Goerlandt, Floris (2017). “A probabilistic model for navigational accident scenarios in the Northern Baltic Sea”. In: DOI: [10.1201/9781315210469-2](https://doi.org/10.1201/9781315210469-2).
- Groeneboom, Piet and G. Jongbloed (Jan. 2014). *Nonparametric estimation under shape constraints: Estimators, algorithms and asymptotics*. Vol. 38, pp. 1–416. DOI: [10.1017/CB09781139020893](https://doi.org/10.1017/CB09781139020893).
- Guillarme, Nicolas Le and Xavier Lerouvreur (2013). “Unsupervised extraction of knowledge from S-AIS data for maritime situational awareness”. In: *Proceedings of the 16th International Conference on Information Fusion*, pp. 2025–2032.
- Harris, Nancy L. et al. (2017). “Using spatial statistics to identify emerging hot spots of forest loss”. In: *Environmental Research Letters* 12.2, 024012, p. 024012. DOI: [10.1088/1748-9326/aa5a2f](https://doi.org/10.1088/1748-9326/aa5a2f).
- IMO (2002). *International Convention for the Safety of Life at Sea (SOLAS), Chapter V: Safety of Navigation, Regulation 19*.
- (2014). *DIRECTIVE 2002/59/EC OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL*. URL: <https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:02002L0059-20110316&from=EN> (visited on 03/15/2019).
- James, Gareth et al. (2013). *An introduction to Statistical Learning*. New York: Springer.
- Kowalska, Kira and Leto Peel (2012). “Maritime anomaly detection using Gaussian Process active learning”. In: *2012 15th International Conference on Information Fusion*, pp. 1164–1171.
- Marsh, Duncan and Duncan L. Marshall (1999). *Applied Geometry for Computer Graphics*. 1st. Berlin, Heidelberg: Springer-Verlag. ISBN: 1852330805.
- Moosavi, Sobhan, Rajiv Ramnath, and Arnab Nandi (2016). “Discovery of driving patterns by trajectory segmentation”. In: *SIGSPATIAL PhD Symposium*.
- Moosavi, Sobhan et al. (2017). “Characterizing Driving Context from Driver Behavior”. In: *SIGSPATIAL/GIS*.
- Nandana, G. M., S. Mala, and A. Rawat (2019). “Hotspot Detection of Dengue Fever Outbreaks Using DBSCAN Algorithm”. In: *2019 9th International Conference on Cloud Computing, Data Science Engineering (Confluence)*, pp. 158–161. DOI: [10.1109/CONFLUENCE.2019.8776916](https://doi.org/10.1109/CONFLUENCE.2019.8776916).

- Pallotta, Giuliana, Michele Vespe, and Karna Bryan (2013). "Vessel Pattern Knowledge Discovery from AIS Data: A Framework for Anomaly Detection and Route Prediction". In: *Entropy* 15, pp. 2218–2245.
- Pieg, Chris (2013). *K Means*. URL: <https://stanford.edu/~cpiech/cs221/handouts/kmeans.html> (visited on 06/02/2019).
- Qin, K. et al. (2017). "HOTSPOTS DETECTION FROM TRAJECTORY DATA BASED ON SPATIOTEMPORAL DATA FIELD CLUSTERING". In: *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XLII-2/W7*, pp. 1319–1325. DOI: [10.5194/isprs-archives-XLII-2-W7-1319-2017](https://doi.org/10.5194/isprs-archives-XLII-2-W7-1319-2017).
- Sitanggang, Imas, A Risal, and Lailan Syaufina (2018). "Incremental Clustering on Hotspot Data as Forest and Land Fires Indicator in Sumatra". In: *IOP Conference Series: Earth and Environmental Science* 187, p. 012043. DOI: [10.1088/1755-1315/187/1/012043](https://doi.org/10.1088/1755-1315/187/1/012043).
- Wood, Frank (2009). *Principal Component Analysis*. URL: <http://www.stat.columbia.edu/~fwood/Teaching/w4315/Fall2009/pca.pdf> (visited on 06/10/2019).
- Yu, X. et al. (2014). "Explore Hot Spots of City Based on DBSCAN Algorithm". In: *2014 International Conference on Audio, Language and Image Processing*, pp. 588–591. DOI: [10.1109/ICALIP.2014.7009862](https://doi.org/10.1109/ICALIP.2014.7009862).