INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA

Área Departamental de Engenharia de Electrónica e Telecomunicações e de Computadores

LEARNING PORTUGUESE FISHING DATA PATTERNS

SERGE GASPAR AGUIAR FERNANDES LAGE

Master degree

Projecto Final para obtenção do Grau de Mestre
em Engenharia Informática e de Computadores

Orientadores : Prof. Doutora Iola Maria Silvério Pinto
Prof. Doutor João Carlos Amaro Ferreira

Júri:

Presidente: [Grau e Nome do presidente do juri]

Vogal: [Prof. Doutor Artur Jorge Ferreira]

March, 2020

INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA

Área Departamental de Engenharia de Electrónica e Telecomunicações e de Computadores

LEARNING PORTUGUESE FISHING DATA PATTERNS

SERGE GASPAR AGUIAR FERNANDES LAGE

Master degree

Projecto Final para obtenção do Grau de Mestre
em Engenharia Informática e de Computadores

Orientadores : Prof. Doutora Iola Maria Silvério Pinto
Prof. Doutor João Carlos Amaro Ferreira

Júri:

Presidente: [Grau e Nome do presidente do juri]

Vogal: [Prof. Doutor Artur Jorge Ferreira]

March, 2020

To my son and wife.

Acknowledgments

I would first like to thank my thesis supervisor, Professor Iola Maria Silvério Pinto for her support, guidance, and oversight along the writing of this dissertation.

I would like to thank Professor João Carlos Amaro Ferreira and the company Xsealence for providing real VMS data from Portuguese vessels.

I would also like to acknowledge my friend Bruno Miguel Carvalhido Lima from Faculdade de Engenharia da Universidade do Porto for his greatly appreciated comments and suggestions.

Finally, I cannot pass up the opportunity to thank my family — my wife in particular —, for her support and continuous encouragement throughout my years of study and through the process of researching and writing this dissertation. This accomplishment would not have been possible without them.

Thank you.

Serge Lage

Contents

List of Figures

List of Tables

1

Introduction

This chapter introduces the motivation, context and the goals of this work. Finally, it presents the overall structure of this document.

1.1 Motivation

1.1.1 Fishing Activity

The increased fishing activities mankind imposed on the marine ecosystems is a threat for the future sea economy and for the marine ecosystem's integrity [1].
Fisheries mapping is needed for implementing better ecosystem management and to secure a healthy marine population [7]. The fishing activity represents an important activity for the Portuguese economy. In the European Union, Portugal is the country that has the highest consumption of fish per person and the third worldwide needing 55,6 kg (per capita/year), [8]. Portugal is a country connected to the sea by its great coast, all along with its continental territory, almost all the coastal villages have a fishing community. An important issue of concern to the authorities is the occurrence of tax evasion that continues to cause damage to the Portuguese economy. In Portugal, the estimated tax evasion in all economic activities represents 21,9% of it's Gross Domestic Product (GDP) [29]. The use of statistical pattern recognition techniques to analyze the data makes possible to identify who operates in the margin of the law more rapidly and methodically [36]. Reducing tax evasion will allow strong gains and potentially

will develop the economy, making the fishing activity more just for everyone involved in this activity.

MONICAP [43] is a monitoring system for the inspection of fishing using the Global Positioning System (GPS) for vessel location and Inmarsat-C [15] technology for satellite communications between ships and a ground control center. MONICAP was successfully introduced on the market by Xsealence [44]) and is currently installed or currently being installed on about 800 fishing vessels operating under the control of the authorities of Portugal, Spain, France, Ireland, and Angola. Within the scope of this Master's thesis, it is proposed to use Portuguese fishing data from the Vessel Monitoring Systems (VMS) to extract patterns of behavior related to the fishing zones, times, speeds, and directions of the course performed by the ships. The descriptive statistical analysis of these makes it possible to identify patterns of fishing activity that can be used for different proposes like sustainable fishing, models for fuel efficiency, and models to detect illegal activities.

VMS provides a unique and independent method to derive patterns of spatially and temporally explicit fisheries activity. Such information may feed into ecosystem management plans seeking to achieve sustainable fisheries while minimizing potential risk to non-target species (e.g. cetaceans, seabirds, and elasmobranchs) and habitats of conservation concern. With multilateral collaboration, VMS technologies may offer an essential solution to quantifying and managing ecosystem disturbance, particularly on the high-seas.

1.1.2 Analytics

The concept of Analytics refers to the ability to use data, perform predictive analytics, and systematic reasoning to improve performance in key business domains and lead to a more efficient decision-making process. There is extensive use of mathematics and statistics, like descriptive techniques and predictive models, which allow gaining valuable knowledge from the data. The insights from data are used to recommend action or to guide decision making rooted in a business context.

1.1.3 Data mining

Data mining is the process of discovering actionable information from large sets of data. Data mining uses mathematical analysis to derive patterns and trends that exist in data. Typically, these patterns cannot be discovered by traditional data exploration because the relationships are too complex or because there is too much data.

1.2 Goals

Objective 1: Local tool
In real-time, will be developed an application to be installed in the MONICAP, which allows better describing the fishing zones. At each one of the vessels, this application will work in real-time with the data from the fishing activity of this vessel. This tool will be local and will be used unsupervised techniques of machine learning. The derived application should be able to identify patterns in two strands: Speed: identify if the vessel is in fishing activity or not; Location: identify the usual fishing spots. These spots knowledge to cross-check in real-time if the current location of fishing is new to the vessel.

Objective 2: Centralized tool
Using VMS data and information on fishing licenses per vessel, our goal is to design models that are capable of classifying vessels by type of fishing only using VMS data. These models could be used to classify vessels by fishing activity, allowing crossing this classification with the information corresponding to the vessel license.

1.3 Document Structure

This document is divided into six main chapters. Chapter 2 gives an overview of the State of the Art concerning the usage of technologies in the fishing industry to detect outliers' behaviors such as activity. In Chapter 3, we focus on the data description and analysis, as well as the needed data pre-processing treatment to find possible solutions for the goals described previously. The data are divided into two categories:

- VMS Records: data generated by MONICAP system;
- VMS Vessels: data coming from the vessel's captains, manually inserted.

In Chapter 4 were presented two methodology's to classify the data in real-time, only using the available data by the Blue Box:

- Using velocity data, to classify if the vessel is fishing;
- Using location data, to classify if the vessel is in activity in a new area;

In chapter 5 is presented an approach to answer the second objective: using the data from all the vessels, how to identify fishing activities that are not under the vessel's

fishing license. Different data mining methods will be used to derive predictive models. Corresponding results are compared through correct classification performance measures. Chapter 6 is presented with the results of the validation of the models and methods presented to answer the proposed objectives. Chapter 7 contains the conclusions obtained during the elaboration of this work.

1.4 Publications

1.4.1 U. C. report to the final project of course (FPC)

My FPC entitled "Análise de Padrões para Encontrar Fraude nas Pescas" was developed in the same data analysis context. In that work I tried to solve an analogous problem with data coming from the VMS file, but with a different approach.
FPC work was focused on abnormalities regarding the declaration of fish caught, by quantities and type of fish. It was used the data provided by the Capitan with quantities caught per type of fish and used VMS Records data to consider standards, as the time of the year and fishing positions.

1.4.2 Published Paper

Fishing Monitor System Data: A Naïve Bayes Approach
Authors: Serge Lage, Iola Pinto, João Ferreira, Nuno Antunes
Book: Springer, Advances in Intelligent Systems and Computing volume 557
Date: 23 February 2017
DOI: 10.1007/978–3–319–43480–0—57
https://link.springer.com/chapter/10.1007/978–3–319–53480–0—57

2

State of the Art

> There exists a desire amongst the world's fisheries managers to coordinate their efforts so that the world's fish stocks - which recognize no national or regional boundaries - can be saved. (Food and Agriculture Organization of the United Nations, Rome, 1998)

To follow this recommendation, there must have to be an agreement concerning the procedures for implementing VMS. For example, when a South America fisheries manager agrees with a fisheries manager in Europe on VMS performance, security and data formats, it will be possible a vessel operates under the management of both, moving from one fishery to another, within legally and a maximum of transparency. Furthermore, only within such a context, can the two fisheries managers share data on vessel movements and activities, to improve operations on an international scale.
VMS is nowadays a standard tool of fisheries monitoring and control worldwide, but it was the EU that led the way, becoming the first part of the world to introduce compulsory VMS tracking for all the larger boats in its fleet. The EU legislation requires that all coastal EU countries should set up systems that are compatible with each other so that countries can share data and the Commission can monitor the respect of the rules. EU funding is available for the Member States to acquire state-of-the-art equipment and to train their people to use it. [9] If an international standard exists, the fisheries managers from all regions of the world would be able to set a common goal. However, there exists some consensus on VMS implementation, providing some welcome, but it will be temporary. This may not be enough to keep everyone on the same track but

could be enough to keep them moving in the same direction.

There is some work being done using VMS data to reach very different objectives like:

- Illegal fishing: "Fishing Gear Recognition from VMS data to Identify Illegal Fishing Activities in Indonesia", [22];
- Fuel efficiency: "Effects of fishing effort allocation scenarios on energy efficiency and profitability: An individual-based model applied to Danish fisheries",[2];
- Sustainable fishing: "The importance of scale for fishing impact estimations",[28];

In terms of tools developed to analyze VMS data, we have two applications (VMStools and VMSbase).

- VMStools: is a package of open-source software, build using the freeware environment R, specifically developed for the processing, analysis, and visualization of landings (logbooks with information of the caught fish) and vessel location data (VMS) from commercial fisheries. Embedded functionality handles erroneous data point detection and removal, linking logbook and VMS data together to distinguish fishing from other activities, provide high-resolution maps of both fishing effort and landings, interpolate vessel tracks, calculate indicators of fishing impact as listed under the Data Collection Framework at different Spatio-temporal scales [24].
- VMSbase: is an R package derived to manage, process, and visualize information about fishing vessel activity (provided by the vessel monitoring system - VMS) and catches/landings (as reported in the logbooks). Standard analyses comprise: 1) tier identification (using a modified CLARA clustering approach on Logbook data or Artificial Neural Networks on VMS data); 2) linkage between VMS and Logbook records, with the former organized into fishing trips; 3)discrimination between steaming and fishing points; 4) computation of spatial effort concerning user-selected grids; 5)calculation of standard fishing effort indicators within Data Collection Framework; 6) a variety of mapping tools, including an interface for Google viewer; 7) estimation of trawled area[34].

The main difference between this work, and this previously mentioned is that they combine VMS data with the logbooks (data of the type of fish captured and quantity). In this work, it will only be used VMS data. The main advantage is that VMS data is

less subject to malicious changes that logbooks taking into account that logbooks are filled by the shipowner. So they are subject to misrepresentation of the truth. VMS data is generated automatically in a closed system like a black box.

3

Data Analysis

This chapter provides information on data and analysis methodologies.

3.1 VMS Records

VMS Data provided by the Xsealence [44] enterprise contained data generated by the MONICAP [43] "Blue Box". Information about the localization, direction, and velocity of the vessel, every 10 minutes is saved in a local database. VMS datasets contained a vessel identification code, a timestamp, the latitude and longitude positions, the speed and the direction. In this dataset, there are 769930 entries from thirty-eight vessels, between 2008-10-30 and 2016-11-04. These data are from vessels operating in the Portuguese shore. This dataset is created automatically by the MONICAP system and follows the concept of integrity and confidentiality.
The variables registered in the dataset are:

- VesselID: Vessel identification;
- Utc: Date time of the log;
- Gps-id: identification of the GPS in use (0 = GPS with EGNOS, 1 = MiniCs GPS);
- Fix/fix2: types of fix in the GPS:
  - 0 = invalid,

  - 1 = standard: valid, without integrity (without EGNOS),
  - 2 = differential: valid, with integrity (with EGNOS),
  - 3 = integrity: valid with integrity (with EGNOS);

- Lat/Lat2: latitude of GPS primary/secondary (in decimal);
- Lon/Lon2: longitude of GPS primary/secondary (in decimal);
- Cog: Course Over Ground. Varies from 0 to 360 clockwise, being 0, facing north;
- Sog: Speed Over Ground (velocity in knots);

In Table 3.1, we can see the summary of the data used in this project. The number of occurrences is 769930. Some cells of the table are empty due to some measures that do not apply to qualitative data. The P_{25} stands for the 1st Quartile, P_{75} stands for the 3rd Quartile and *SD* stands for Standard deviation.

	Minimum	Maximum	Average	P_{25}	Median	P_{75}	SD
Lon	-52.706	35.965	-4.493	-9.811	-9.115	-7.986	21.156
Lat	-35.243	76.064	25.933	33.038	38.423	40.21	25.916
Sog	0	42	4.183	1.634	3.012	7.399	3.211
Cog	0	360	166.31	68.89	173.125	255.69	108.64
utc	2008-10-30	2016-11-04	-	-	-	-	-
gps_ id	0	1	-	-	-	-	-
fix	1	3	-	-	-	-	-
Fix2	0	2	-	-	-	-	-
Lon2	-52.706	155.977	-3.702	-9.726	-8.991	0	20.933
Lat2	-35.24	76.064	23.663	0	37.595	40.191	26.383
VesselId	1	38	-	-	-	-	-

Table 3.1: VMS Records

3.2 VMS Vessels

VMS Vessels data is the vessel information that goes along with the VMS Records. These data contain information about vessels and fishing activities for which they are licensed. This data is created by the competent authority that process fisheries licensing.
The variables registered in the dataset are:

- ID: Vessel identification (VesselID/VMSRecords, foreign key);
- Name: Name of the vessel;
- Loa: Length Overall;
- GT: Gross Tonnage;
- HP: Vessel power (HP);
- kW: Vessel power (KW);
- License: Registration of the vessel's licenses;
- PriGearCode: FOA code of the principal fishery device;
- SecGearCode: FOA code of the secondary fishery device.

In Table 3.2 we can see the summary of the data regarding the vessels in the table VMS Records. This data was filtered from 56 occurrences to 38. The deleted data was referred to Vessels that have not data in the table VMS Records, so this vessel information was superfluous to our needs. In Appendix A we discriminate the fishing licenses.

	Minimum	Maximum	Average	P_{25}	Median	P_{75}	SD
ID	1	38	-	-	-	-	-
Name	-	-	-	-	-	-	-
Loa	11.95	84.94	23.48	16.93	19.35	23.70	15.49
GT	22251	18.99	200.28	27.98	57.15	110.34	473.78
HP	3600	130	539	230	350	497	689.56
KW	2684.50	95.62	396.52	172.84	259.21	367.91	498.54
License	-	-	-	-	-	-	-
PriGC	-	-	-	-	-	-	-
SecGC	-	-	-	-	-	-	-

Table 3.2: VMS Vessels

3.3 Descriptive Data Analysis

The data used as input to the models is VMS Records data. Among these data, those that bet distinguish between different types of fishing activity are speed and location. About the locations, certain fishing types only occur in certain depth. So the locations can help in these cases.
To meet the objectives, we need to understand the velocity patterns. By studying and analyzing the velocity data, it was possible to verify the existence of two velocity distributions.

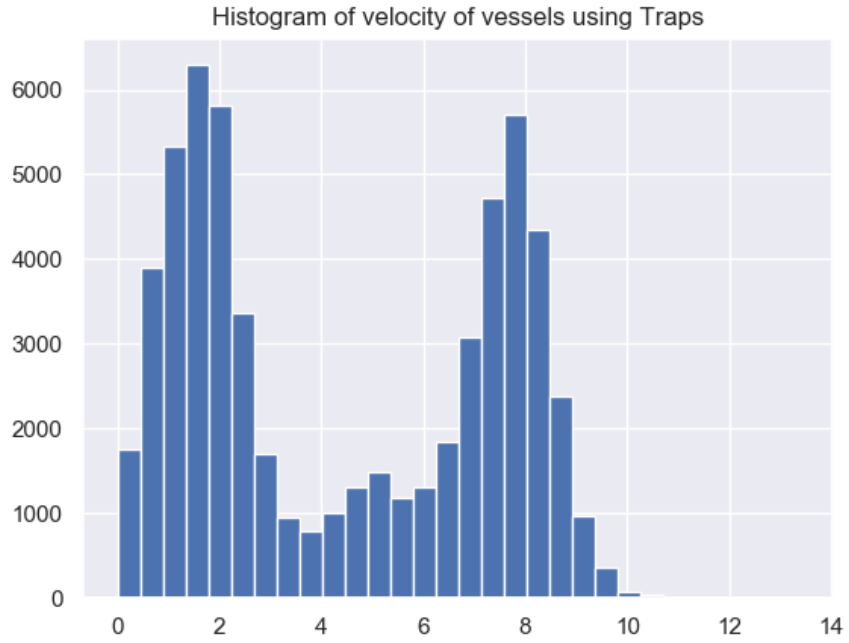

Figure 3.1: Histogram of velocity of traps license.

In Figure 3.1 it is possible to distinguish two different speed distributions, the lower speeds correspond to fishing activities, and the higher speeds represent the movements of the vessel from the port to the fishing grounds and back to the port [11].

Knowing this, we can notice that different types of fishing have different fishing speeds, as we can see the differences between the histogram in Figure 3.1 containing data on fishing vessels using traps and Figure 3.2 concerning fishing vessels using fishhook.

It is essential to separate the inputs representing fishing speeds from the others in the data as the input to the models used in Chapter 5 it is necessary to have the data filtered out only with the fishing activity data. This because we will only use the fishing speed to study and create the methodology's necessary to achieve the proposed goals.

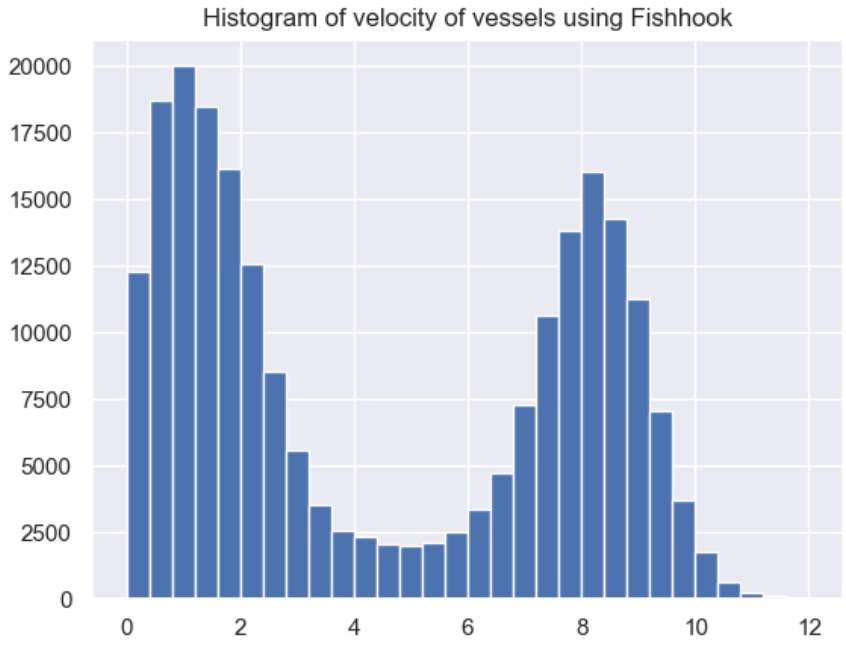

Figure 3.2: Histogram of velocity of fishhook license.

Data from operations other than fisheries should be discarded as they do not discriminate against different types of fishing. For example, speed 0 nautical miles is not interesting regardless of their type of fishing, once all vessels stop in the fishing port.

4

Standalone Fishery Analysis

This chapter explains the approach used to reach the first goal of this work. It describes an application that implements the work done in this chapter with respect to its functionality, architecture, implementation details and usage.

The first objective is to develop a locally implemented tool that registers whether the vessel is engaged in fishing, and if so, whether the fishing area is new or is habitual. This solution must be implemented by the vessel. Therefore each vessel will only have access to its data. That is, each vessel only will know it is one data. This data consists of VMS data described in Section 3.1 VMS Records.

The solution developed to meet this objective consists of a machine learning application to analyze data in real-time, to determine whether the vessel is fishing, and if so, whether it is fishing in its fishing zone or in new location. The fact that this analysis is done by vessel allows avoiding bias in results, since each vessel has a different power, size and its suitable for specific a fishing activity.
This solution could be implemented and used as a library by the MONICAP system shown in Figure 4.1 .[43].

Figure 4.1: MONICAP Blue Box.

As MONICAP systems are installed on ships, they can, in real-time, send alerts to the authorities whenever an abnormal change is detected concerning the standard.

4.1 Fishing Velocity Patterns

To know whether a vessel is fishing, we can use it is velocity patterns, given that the speed of the vessel differs where it is traveling or when it is fishing. We can verify this fact in the histogram shown in Figure 4.2, corresponding to a vessel velocity.

In Figure 4.2, the histogram allows us to recognize two different velocity patterns, identified by two distinct distributions. They are visible when we graphically represent the velocity's data of each of the vessels. The distribution characterized by lower average speeds corresponds to fishing activity, and the other speed distribution corresponds to the movement of the vessel between the port and the fishing sites [11].
So, it is needed to isolate the first distribution's range to be able to classify the upcoming future velocity's as fishing associated or not.

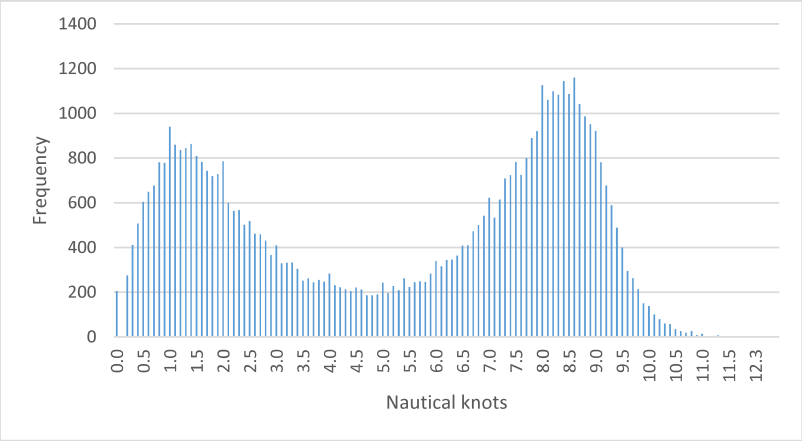

Figure 4.2: SOG Histogram vessel 2.

For the propose of isolating the first distribution was considered and analyzed the potential of three different procedures:

- **Standard deviation:** This solution assumes that the velocity distribution is normal. Using the distribution of the fishing velocity, we find the fishing velocity with more occurrences and with the standard deviation, we can choose a distance from the mean. So we can get the minimum population desired (explained by Chebyshev's inequality [35]) to be within the fishing range.
  Pros and cons: This solution is simple and fast to implement, but we are not working with a normal distribution.

- **Kernel Density Estimation:** Kernel Density Estimation method estimates the probability density function by imposing a model function on every data point and then adding them together. The function applied to each data point is called a kernel function [21]. Pros and cons: This solution gives us a way to configure the threshold for the fishing distribution limits. This solution will not work because the density distribution for the velocity will differ from vessel to vessel.

- **Filter:** Use a Hill-Climbing algorithm[20]. This algorithm is a local optimization algorithm which provides a direct search. The Stochastic Hill-Climbing algorithm works supported in an iterate process of randomly selecting a neighbor for a candidate solution. The acceptance of the solution is conditioned by a criterion of improvement concerning the previous solution. With this algorithm, the first maximum is found, and then the algorithm identifies the next minimum. Then remove all the velocity occurrences that happen less than 10 % of the maximum

occurrence and isolating the occurrences that are followed. A clean distribution of the fishery speed for each vessel is derived. With this, the minimum and the maximum values of this distribution are used to classify the new inputs. Pros and cons: Solution that is independent of fishing type distribution. Needs to organize the velocity data for the Hill-Climbing algorithm. Create a version that finds the local maximum and then finds the next minimum.

After to experiment and study all these different methods, the chosen procedure can be described in two steps: it starts by using the method based on the **Filter** to isolate the fishery speed from the remain. Then the Kernel distribution method was applied.

1. **Filter:** In the first step, it retrieves all velocity data from the database to create a histogram like it is shown in Figure 4.2. In the next step, it uses the hill-climbing algorithm to get the minimum and the maximum value of the first distribution. The implementation used was altered in a way that when the algorithm converges to the maximum, it will continue to find the limit of the distribution. To obtain this solution, the algorithm searches for the first local maximum that does not have a higher value in the following three points. In this way, we can find the maximum value of the fishing speed range.
   To find the end of the fishing speed range, the algorithm continues to sweep the histogram until the next three points are not lower than the current point. Velocity 0 is removed because we do not want to consider when the vessel is completely stopped. This way, we can end up with a histogram of the intended distribution, as we can observe in Figure 4.3.

2. **Kernel:** It was applied a kernel distribution method in the filtered histogram to have the distribution represented in orange on 4.4. Then it was created a dictionary with the velocities and the cumulative percentage of velocity. This way, we end up with a representation like the one presented in Figure 4.5. Then a range across quantiles is defined for some probability. Considering this last distribution, a confidence area is defined throught a probability. The speeds within this area corresponds to fishing activity. Thus two-speed limits are identified and used classify the new data.

Now, we can compare the new data with the established limits. If the new data is within limits, we classify as fishing, and if not, we classify as not fishing.

Appendix B has presented the functionality of this algorithm, step by step.

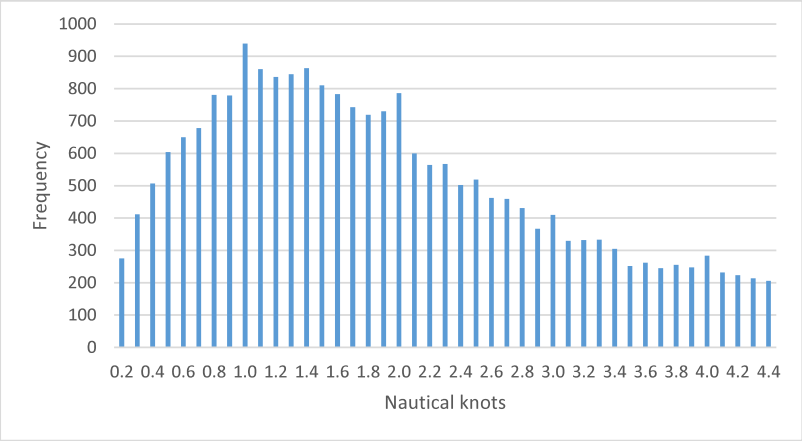

Figure 4.3: Velocity distribution of vessel 2 after the application to the Hill Climbing algorithm

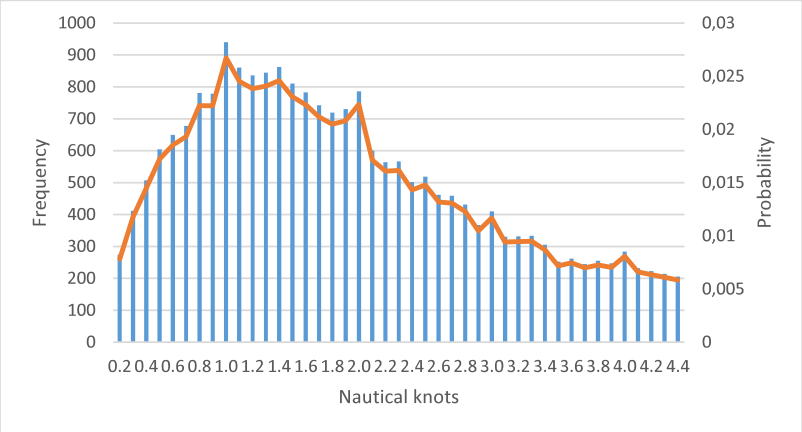

Figure 4.4: Kernel density distribution filtered data

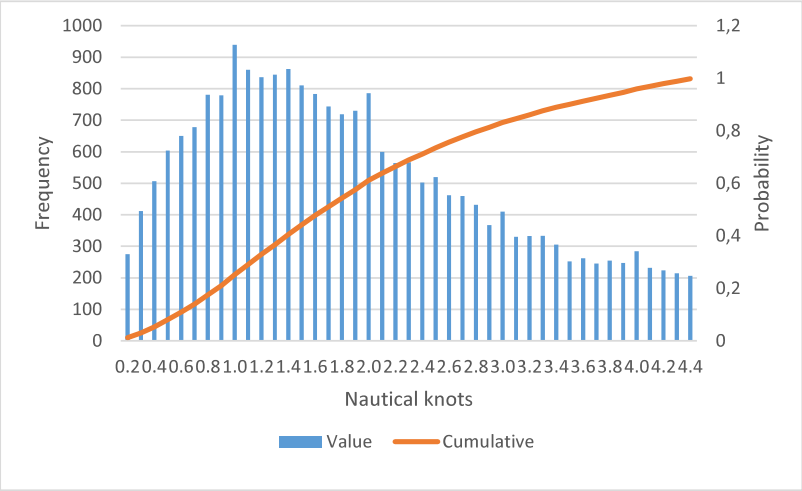

Figure 4.5: Filtred histogram with cumulative kernel distribution

4.2 Fishing Spots

To discover whether the the vessel is fishing in its fishing zone, or in a new location, the history of GPS locations by vessel was used. Fishing in a new zone may mean that the vessel has changed its type of fishing or is engaging in an activity that is not licensed.

In Figure 4.6, we can see the GPS points for the vessel 2. Using methods based on clustering, it is possible to identify, by vessel, several areas that are the standard fishing zones of this vessel. When the vessel is outside that standard zone, a flag should occur.

Using the fishing velocity range encountered in the previous point, we get the GPS points of the vessel within that range, so we can work only with the positions where the vessel was fishing. The next step is to use a clustering algorithm to define the fishing areas so that we can compare it with the new GPS points.

For this purpose, several data mining algorithms were performed in order to choose the best results:

- **K-Means:** K-means clustering algorithm [12] is a method of cluster analysis which aims the partition of n observations into k clusters, in which each observation belongs to the cluster with the nearest mean. This results in a partitioning of the data space. K-means (Macqueen, 1967) is one of the simplest unsupervised learning algorithms that solve the well-known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assuming k clusters) fixed a priori. The main idea is to define k centroids, one for each cluster. These centroids should be placed cunningly

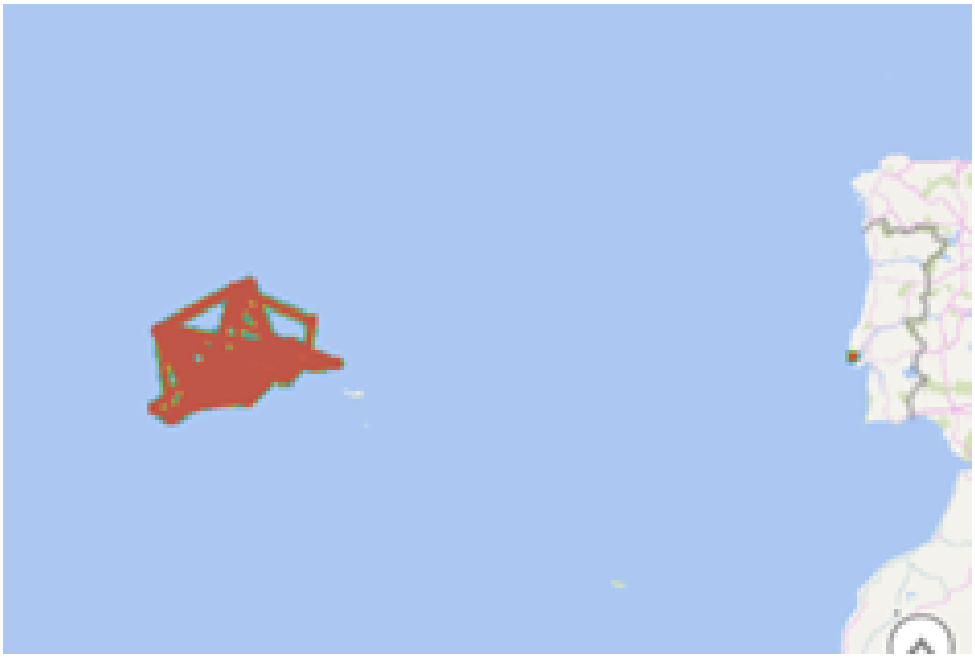

Figure 4.6: Vessel 2 GPS points

because of different location causes a different result. So, the better choice is to place them as much as possible, far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed, and an early group is done. At this point, we need to recalculate k new centroids as bar centers of the clusters resulting from the previous step. After we have these k new centroids, a new binding must be done between the same data set points and the nearest new centroid. A loop has been generated. As a result of this loop, we may notice that the k centroids change their location, step by step, until no more changes are done.

- **Density Based Cluster:** Density-based clustering algorithms [38] try to find clusters based on the estimation of the density of data points in a region. It can find arbitrarily shaped clusters, and handles noises, and yet is a one-scan algorithm that needs to examine the raw data only once. In density-based clustering algorithms, dense areas of objects in the data space are considered as clusters, which are segregated by low-density area (noise). The basic idea of density-based clustering is that clusters are dense regions in the data space, separated by regions of lower object density [39]. The key idea of density-based clustering is that for each instance of a cluster, the neighborhood of a given radius (Eps) must contain at least a minimum number of instances (Min Pts).

	K-Means	Density Based Cluster	DBSCAN
Initializing	862	923	25848
New data	25	45	35

Table 4.1: Time of processing in miliseconds per model

- **DBSCAN:** DBSCAN (for density-based spatial clustering of applications with noise) is a data clustering algorithm proposed by Martin Ester, Hans-Peter Kriegel, Jorge Sander and Xiaowei Xu in 1996. It is a density-based clustering algorithm because it finds several clusters starting from the estimated density distribution of corresponding nodes. DBSCAN [17] is one of the most common clustering algorithms and most cited in the scientific literature.

After some tests, it was decided that Density-Based Cluster is the best approach for this case. It was excluded DBSCAN, because as we can observe in Table 4.2, this model needs much processing power to estimate the clusters. These values were retrieved using a computer with an Intel i5 (2.5 GHz) and 8 GB of RAM. Considering that the Blue Box as a lot less processing power, it was decided that this model is not a good solution for this problem.

The choice between K-Means and Density-Based Cluster algorithms was based on the fact that Density-Based Cluster represents a great advantage because it estimates the probability of the new GPS point belonging to a cluster-based in the specific cluster probabilistic distribution. This way, the user can choose the most suitable configuration. The resulting clusters themselves are equal when K-Means or Density-Based Cluster were applied since Density-Based Cluster uses K-Means to define the centroids, so they only differ by adding a layer to define the area of density per cluster.
To decide the number of clusters to use, it was used the elbow method [18], for the within-cluster sum of squares, as could be seen in Figure 4.7. The within means the distance the vectors in each cluster are from their respected centroid. The goal is to get this number as small as possible. One approach to handling such an objective is to run the K-means clustering multiple times, raising the number of the clusters each time. Then, it is possible to compare the quantity within each time, stopping when the rate of improvement drops off. The better case corresponds to find a low withinss while still keeping the number of clusters low.
The elbow method is visual. The idea is to start with K=2 and keep increasing it in each step by one unit, calculating the clusters and the cost that comes with the training. At some value for K, the cost drops dramatically, and after that, it reaches a plateau when

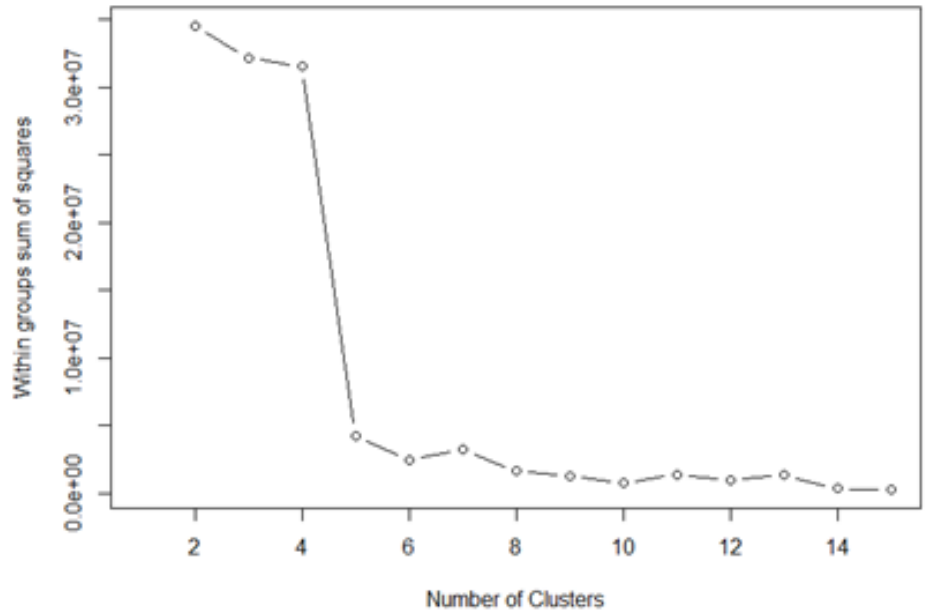

Figure 4.7: Sum of squared error by number of clusters

you increase it further. At this moment, the value of K we are looking for is reached. In Figure 4.7 were the GPS points of all vessels. As we can observe that six clusters are a good number as the error is not decreasing much as the number of clusters increases.

4.3 SFA Library

It was created a software application called SFALib for (Standalone Fishery Analysis Library. In this application, were developed the solutions described in this chapter to help with the elaboration and tests for this project. For a market solution, this library could be used by the main application of a Blue Box, to send alerts to support decision making or to simply classify each VMS data entry into two categories:

1. Is fishing (yes/no).

2. Is fishing in a new area (yes/no).

4.3.1 Functionality

This application allows to:

- Test new data
  Send VMS data and receive it if it is considered to be fishing and if it is in a new area.

- Test new velocity
  Send sog data and receive true if it is considered to be fishing.

- Test new location
  Send GPS data and receive true it is in a new area.

- Restart models
  Request to create new models. It can be used if the objective is running for a long time and want to renew models with new data.

- Get limits
  Request the velocity limits. Receive a tuple with two doubles (item1 = low-speed limit, item2 = high-speed limit). It can be used for analysis like is described in Chapter 5.

To make this possible is necessary to configure the data access layer to get the VMS data from the local data repository. Currently, the application supports conection to SQL Server [23] and PostgreSQL [30].

4.3.2 Architecture and Implementation

To develop the software, it was decided to use Java 8 [26] because it is a powerful, full object-oriented, and cross-platform programming language. MONICAP uses Linux, so using a JRE (Java Runtime Environment) application is a good choice. The architecture is depicted in Figure 4.8. In this architecture is possible to distinguish three main modules: One that is the core of the SFALib, create the modules and use them. Another one is WEKA [40] that creates the cluster modules for locations and implements the Kernel density estimation for velocity. The last one is the data access layer that is responsible for getting the VMS data from the local repository so SFALib can create the modules.

Core module
The core module is responsible for initializing the models and using them the way described in this chapter.The procedure starts with the creation of an instance, called "ProcessVelocity", whose objective is to use the historical speed data of the ship to

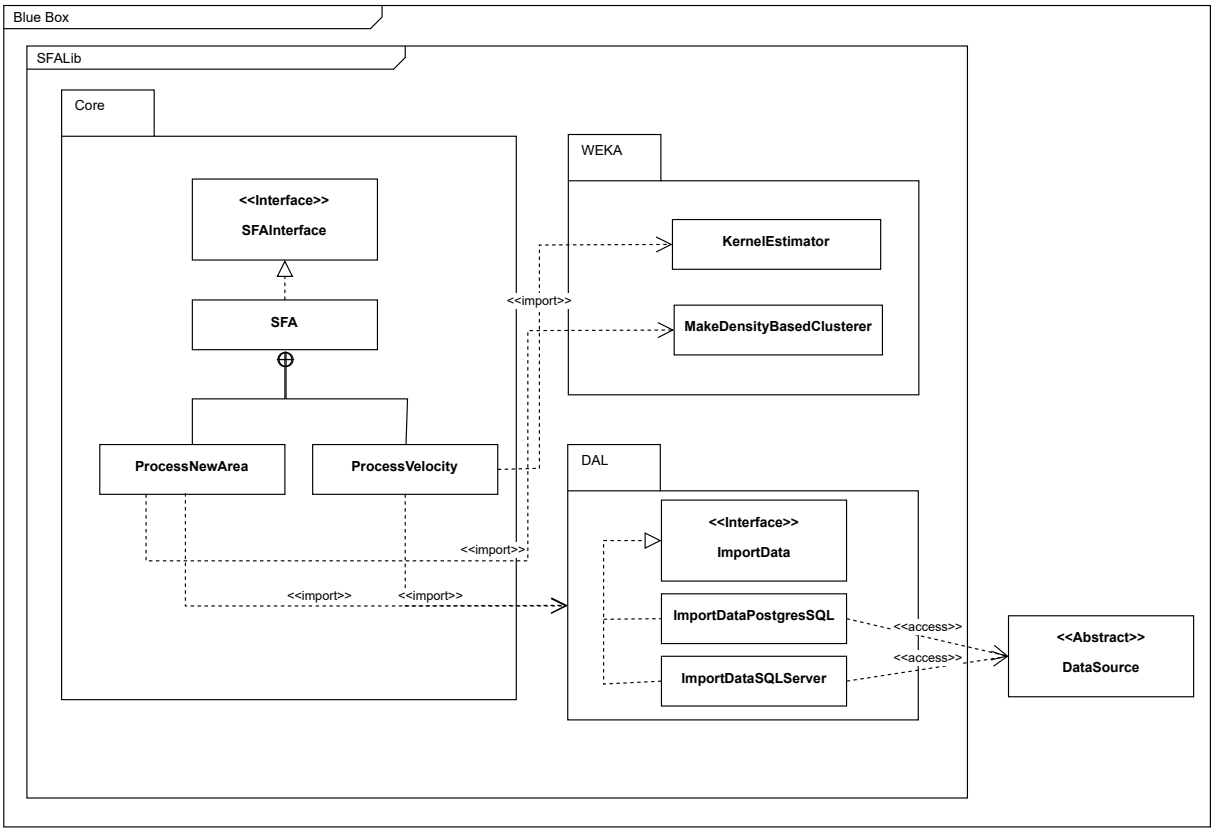

Figure 4.8: Representation of the SFALib architecture.

classify whether or not it is fishing. After that is created another instance called, "ProcessNewArea", whose objective is to identify if a new data (GPS point) is or is not in a usual fishing location for that vessel, using for that purpose the history of the vessel's GPS locations.

WEKA module
WEKA is a collection of machine learning algorithms for data mining tasks. It contains tools for data preparation, classification, regression, clustering, association rules mining, and visualization. In this application, WEKA is used as a tool to create the modules.

DAL module
The data access module was implemented in a way not only to get data but also to filter data in the database engine. Filtering data (where) is optimized on the database engine, and so we gain some performance.

The application starts by initializing two objects:

1. ProcessVelocity: This object is responsible for doing the process explained in point 4.2 of this document. This object will request to the static class ImportData to retrieve all SOG (Speed Over Ground) data from the database. Then the process will end with the limits (minimum speed of fishing and the maximal speed of fishing).

2. ProcessNewArea: This object is responsible for doing the process explained in point 4.3 of this document. This object is only initialized after ProcessVelocity because it needs the velocity fishing limits necessary to create the clusters of the fishing areas. With these limits, the object requests to the instance ImportData to obtain the latitude and longitude values where the vessel was in between the velocity limits. With this, the object ends up with the clusters of the fishing areas.

4.3.3 Deployment

We start by initializing SFALib with the doubles limitVelocity and limitArea. This doubles range between 0 and 1:

- limitVelocity: used to get the maximum and minimum speed by reducing the speed range. This limit will reduce de maximum speed and increase the minimum speed by setting the maximum velocity as the velocity that as (1-limit) percentage of the cumulative kernel distribution and the minimum velocity as the velocity that as a (limit) percentage of the cumulative kernel distribution.

- limitArea: used to compare with the probability to belong in a cluster given to the new points. If the limit is smaller than the given probability, then the vessel is classified as fishing in a new area.

These limits could be defined by the user. This possibility allows to configure the application according the preference in obtaining more false positive or false negative classifications. A false positive (type I error) is when the classifier rejects a true hypothesis. A false negative (type II error) is when the classifier accepts a false hypothesis.

After we SFALib is ready, we need to send a new velocity data and GPS coordinates to receive an object with an "isFishing" as true if the vessel is fishing. "isNewArea" as true if the vessel is in an area that is not a normal fishing area and it's in a fishing velocity.

The methods that can be used are:

- newData method that receive VMS data and return isFishing(boolean) and isNewArea(boolean).
- isFishing: method with SOG(double) as input and a boolean as output with, True = is fishing, False = not fishing.
- isNewArea: method with GPS(double longitude, double latitude) as input and a boolean as output with, True = is fishing in a new area, False = not fishing in a new area.
- restart: this method restart the models. It can be used to create models with new data.
- GetLimit: the method used to get SOG limits used by the SFALib to classify velocity. As Limits(double min, double max) as output.

5

Joined Fishery Analysis

This chapter explains the approach used to reach the second goal of this work.

Data mining is the process of discovering interesting and useful patterns and relationships in large volumes of data. The field combines tools from statistics and artificial intelligence (such as neural networks and machine learning) with database management to analyze extensive digital collections, known as data sets. Data mining is widely used in business (insurance, banking, retail), science research (astronomy, medicine), and government security (detection of criminals and terrorists) [25].

5.1 CRISP-DM

In this work, it will be used the CRISP-DM (Cross Industry Standard Process for Data Mining) methodology [42]. The CRISP-DM project proposed a comprehensive process model for carrying out data mining projects. The process model is independent of both the industry sector and the technology used [42]. The CRISP-DM reference model for data mining provides an overview of the life cycle of a data mining project. It contains the phases of a project, their respective tasks, and their outputs. The life cycle of a data mining project is broken down into six phases, which are shown in Figure 5.1. The sequence of the phases is not strict. The arrows indicate only the most important and frequent dependencies between phases, but in a particular project, it depends on the outcome of each phase, which phase, or which particular task of a phase, has to be performed next.

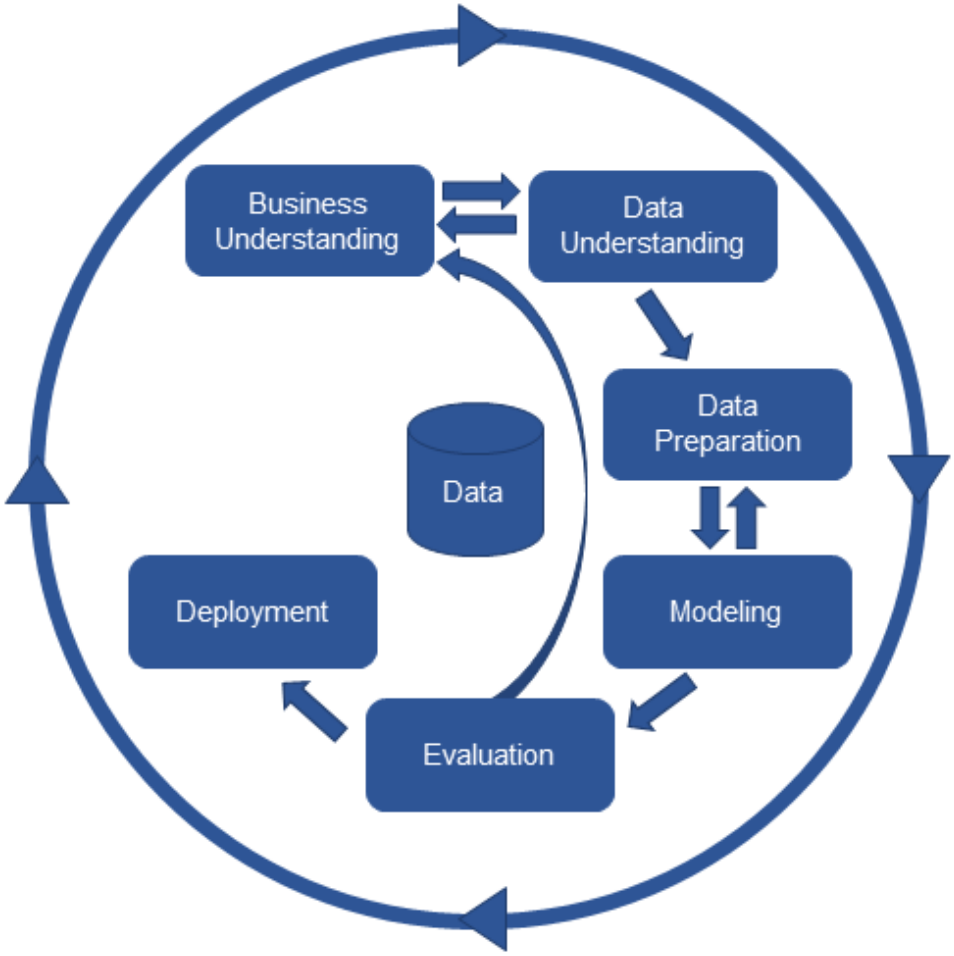

Figure 5.1: Complete CRISP-DM Approach [13].

In the following, we outline each phase briefly:

- Business Understanding
  This initial phase focuses on understanding the project objectives and requirements from a business perspective, and then converting this knowledge into a data mining problem definition, thus a preliminary project plan designed to achieve the objectives are drown.

- Data Understanding
  The data understanding phase starts with an initial data collection and proceeds with activities to get familiar with the data, to identify data quality problems, to discover first insights into the data, or to detect interesting subsets to form hypotheses for hidden information. There is a close link between Business Understanding and Data Understanding. The formulation of the data mining problem and the project plan require at least some understanding of the available data.

- Data Preparation
  The data preparation phase covers all activities to construct the final dataset (data that will be fed into the modeling tool(s)) from the initial raw data. Data preparation tasks are likely to be performed multiple times, and not in any prescribed order. Tasks include a table, record, and attribute selection, data cleaning, construction of new attributes, and transformation of data for modeling tools.

- Modeling
  In this phase, various modeling techniques are selected and applied, and their parameters are calibrated to optimal values. Typically, there are several techniques for the same data mining problem type. Some techniques require specific data formats. There is a close link between Data Preparation and Modeling. Often, one realizes data problems while modeling, or one gets ideas for constructing new data.

- Evaluation
  At this stage in the project, you have built one or more models that appear to have high quality, from a data analysis perspective. Before proceeding to the final deployment of the model, it is essential to more thoroughly evaluate the model, and review the steps executed to construct the model, to be sure it accurately achieves the business objectives. A key objective is to determine if there is some critical business issue that has not been sufficiently considered. At the end of this phase, a decision on the use of the data mining results should be reached.

- Deployment
  The creation of the model is generally not the end of the project. Usually, the knowledge gained will need to be organized and presented in a way that the customer can use it. Depending on the requirements, the deployment phase can be as simple as generating a report or as complex as implementing a repeatable data mining process. In many cases, it will be the user, not the data analyst, who will carry out the deployment steps. In any case, it is important to understand upfront what actions will need to be carried out to make use of the created models.

5.2 Implementation

5.2.1 Business Understanding

In the fishing sector, vessels operating in the various fishing techniques must be licensed. A common problem is that there is a likelihood that vessels will be fishing for which they are not licensed. The objective is to classify VMS data by fishing type. This way, we can try to confirm that each ship is performing a legal activity according to the license previously obtained.

5.2.2 Data Understanding

To answer this goal, the data used is the same VMS Records as used in chapter 4, and VMS Vessels explained in chapter 3.2. Initially, this study will focus on the analysis of velocities. However, the other variables will not be discarded until their contribution to the adequate solution is verified.

5.2.3 Data Preparation

To use data mining models, a first step is to build a dataset with all the data needed to feed the models. So, it was created a dataset from VMS Vessels and VMS Records to end with Table 5.2.3.

Name	Description	From	Why
ID	Key	Native	Identify the row
VesselID	Vessel Identifier	VMS Records	Identify the vessel
UTC	Date Time	VMS Records	Identify the time of the entry
LAT	Latitude	VMS Records	Discriminated by fishing areas
LON	Longitude	VMS Records	Discriminated by fishing areas
COG	Direction	VMS Records	Course Over Ground
SOG	Velocity	VMS Records	Discriminated by fishing velocity
LOA	Length Overall	VMS Vessels	Discriminated by vessel type
GT	Gross Tonnage	VMS Vessels	Discriminated by vessel type
HP	Vessel Power	VMS Vessels	Discriminated by vessel type
License	Vessel's Linceses	VMS Vessels	Objective

Table 5.1: VMS Dataset

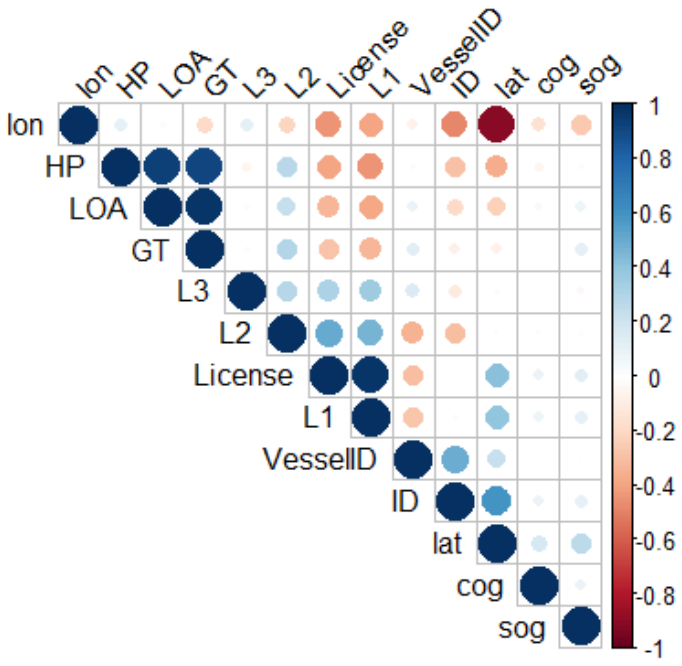

Figure 5.2: Data Correlation

To analyze the existence of associations between the type of license and the remaining variables, Pearson correlation correlation coefficient was obtained for all the pairs of variables. Pearson coefficient [3] measures the degree of correlation and the direction of this correlation - whether positive or negative (between two metric scale variables). In some cases, it was necessary to pre-process the data before calculating the association indicator. For example, it happens in the case of the variables SOG and license. In Figure 5.2, the results of the correlation coefficients were presented.

We can see that we cannot use the data as it is because of the correlation between License and sog is very weak, so we need to do some pre-processing. The best way to achieve a higher correlation between fishing speeds and licenses was the transformation of data to storage fishing average, maximum, and minimum per day per vessel. This way, it presents the results shown in Figure 5.3.

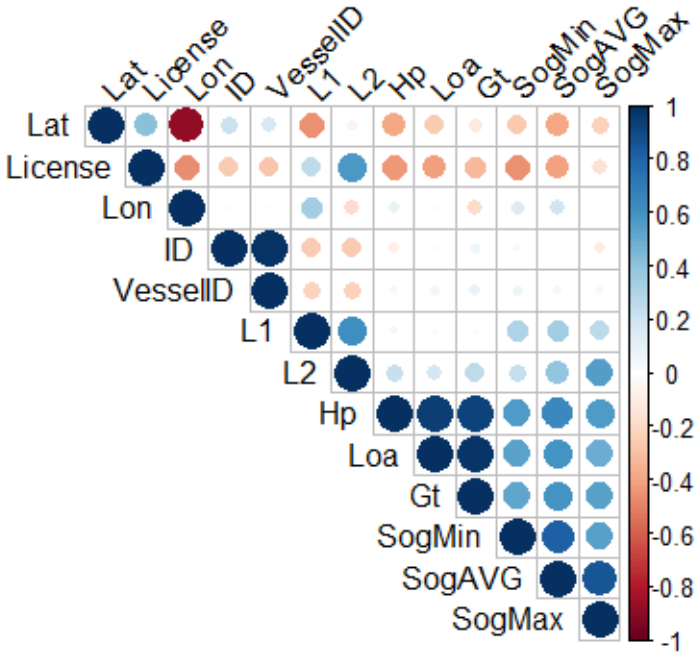

Figure 5.3: Data correlation after preprocessing

The method used to transform the data consists of the following steps:

- Create a dataset in which data are grouped per day, per vessel.
- Use DSALib to get the minimum and maximum speeds of fishing per vessel. From this data, apply a filter to obtain observations for which the velocity varies between its minimum and maximum value.

With this data, we can also observe that the correlation between the license and the HP data (vessel power), Loa (Boat length) Gt (vessel weight) are quite significant. This is expected as different fishing activities require specific types of vessels. This does not mean that the type of vessel is only capable of entering a type of fishing activity. For these reasons, we will not use these variables in the model so as not to create a problem with bias.
Regarding location data, clustering techniques to discretize the data were used. First, the best number of clusters is determined. For that, we used the same technique used in chapter 4.3, resulting the Figure 5.4. The data used was the dataset filtered, so we have only the positions of fishing. The chosen number of clusters was 4.

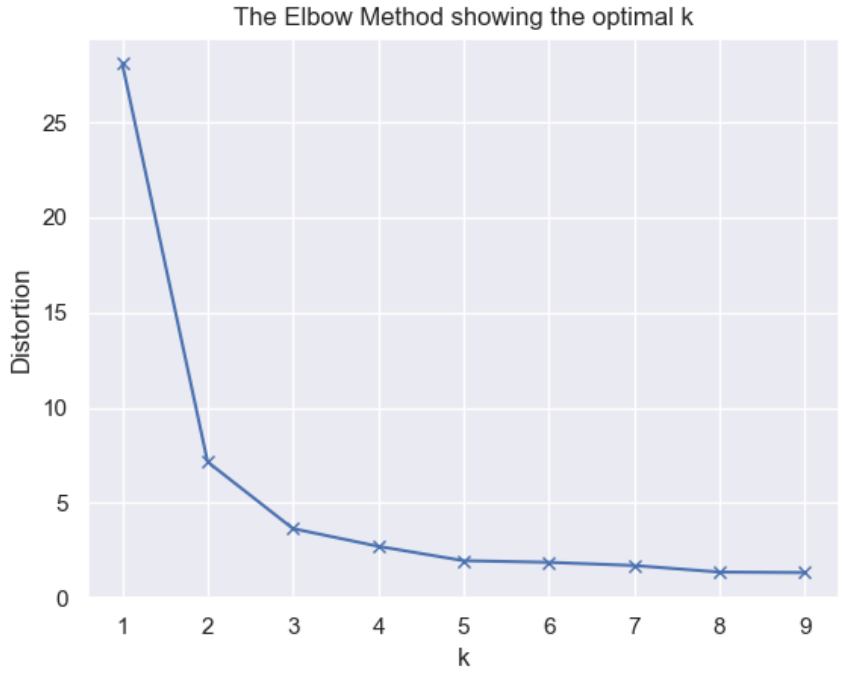

Figure 5.4: The elbow method showing the optimal k

Now, is possible to create data mining models based on the location and on the velocity patterns.

5.2.4 Modeling

In order to classify a new type of license using VMS data, several data mining algorithms were tested, which we now list:

- **KMeans:** This model was used to classify GPS data in clusters to improve the result of the data mining algorithms. The operating mode of this algorithm has already been covered in the 4.3.

- **Decision Trees:** While in data mining, a decision tree is a predictive model that can be used to represent both classifiers and regression models, in operations research decision trees refer to a hierarchical model of decisions and their consequences[33]. The decision-maker employs decision trees to identify the strategy which will most likely reach its goal. When a decision tree is used for classification tasks, it is most commonly referred to as a classification tree. When it is used for regression tasks, it is called a regression tree [33]. Algorithms for constructing decision trees usually work top-down, by choosing a variable at each step that best splits the set of items [32].
  In this work, it will be used to measure the quality of a split "gini" for the Gini impurity (CART)[14] and "entropy" for the information gain (C4.5)[14].

- **Random Forests:** Random forests are a combination of tree predictors such that each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest[4]. The generalization error for forests converges a.s. To a limit as the number of trees in the forest becomes large. The generalization error of a forest of tree classifiers depends on the strength of the individual trees in the forest and the correlation between them. A random forest is a classifier consisting of a collection of tree-structured classifiers $\{h(\mathbf{x}, \theta_k), k = 1, ...\}$ where the $\{\theta_k\}$ are independent identically distributed random vectors and each tree casts a unit vote for the most popular class at input **x**[4].

- **Neural Network:** Neural networks are a bio-inspired mechanism of data processing that enables computers to learn technically similar to a brain and even generalize once solutions to enough problem instances are tough [19]. A technical neural network consists of simple processing units, the neurons, and directed, weighted connections between those neurons. A neural network is a sorted triple (N, V, ϖ) with two sets N, V and a function ϖ, where N is the set of neurons and V a set $\{(i, j)i, j \in \mathbb{N}\}$ whose elements are called connections between neuron i and neuron j. The function $\varpi : V \to \mathbb{R}$ defines the weights, where $\varpi((i, j))$, the

weight of the connection between neuron i and neuron j, is shortened to $\varpi_{i,j}$. Depending on the point of view, it is either undefined or 0 for connections that do not exist in the network [19].
In this work, we will train models with different hidden layer sizes. The solver for weight optimization used is BFGS[6] and Adam[16] for large sizes of hidden layers.

- **Support Vector Machine:** The folklore view of SVM is that they find a optimal hyperplane as the solution to the learning problem. The simplest formulation of SVM is the linear one, where the hyperplane lies in the space of the input data x. In this case, the hypothesis space is a subset of all hyperplanes of the form: $f(x) = w \cdot x + b$. In their most general formulation, SVM finds a hyperplane in a space different from that of the input data x. It is a hyperplane in a feature space induced by a kernel K (the kernel defines a dot product in that space)[10].
In this work, we will create models using the following kernel algorithms: Linear[45], Polynomial[45], and RBF[45].

5.2.5 Evaluation

Cross –validation [37] provides a simple and effective method for both model selection and performance evaluation, widely employed by the machine learning community. Under k –fold cross –validation, the data are randomly partitioned to form k disjoint subsets of approximately equal size. In the ith fold of the cross-validation procedure, the ith subset is used to estimate the generalization performance of a model trained on the remaining k –one subset. The average of the generalization performance observed overall k folds provides an estimate (with a slightly pessimistic bias) of the generalization performance of a model trained on the entire sample. The k used to test these models is 10.
To evaluate the classification results customized with the different algorithms, a confusion matrix will be created and interpreted according ours goals as in the table 5.2.5.

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

Table 5.2: Confusion matrix

In addition, the following performance indicators are used:
Precision = $\sum_{n=0}^{k-1} \frac{tp}{tp+fp}$ = precision of the model.
Accuracy = $\sqrt{precision^2}$ = accuracy of the precision result.

A confusion matrix [27] illustrates the accuracy of the solution to a classification problem. Given n classes, a confusion matrix is a m x n matrix, where $C_{i,j}$ indicates the number of tuples from D that were assigned to class $C_{i,j}$, but where the correct class is C_i. The best solution will have only zero values outside the diagonal. A confusion matrix contains information about actual and predicted classifications done by a classification system. The performance of such systems is commonly evaluated using the data in the matrix. The following table shows the confusion matrix for a two-class classifier.

For the propose of this work, the classes are corresponding to the different types of licenses, are:

0 Armadilhas / De abrigo / Alcatruzes
1 Arrasto / De fundo de portas
2 Arrasto / De fundo de portas / Crustáceos
3 Arrasto / Pelágico / Com portas
4 Cerco / para bordo / Tipo americano
5 Emalhar de 1 pano / De deriva / Grandes Pelágicos
6 Emalhar de 1 pano / De fundo
7 Pesca à linha / Cana e linha de mão
8 Pesca à linha / Palangre de fundo / Espécies demersais
9 Pesca à linha / Palangre de Fundo + Cana e linha de mão
10 Pesca à linha / Palangre de superfície / Grandes Migradores

The model's test results are:

- **Decision Trees:**

  We can observe in Table 5.2.5 the usage of velocity parameters (SogMin, SogAVG, and SogMax) and location (clustering result of K-Means) have the best result. The algorithm with the best result is Entropy(C4.5), with a precision of 0.7776 and an accuracy of 0.1459. In Figure 5.2.5 the confusion matrix shows that only the classes 0 and 7 have a low prediction rate. The max depth of the trees was tested as 200, 300, and 400, with 300 giving the best results.

Algorithm	**Velocity and locations**		**Velocity**	
	Precision	Accuracy	Precision	Accuracy
Gini	0.774	0.1481	0.7236	0.1349
Entropy	0.7776	0.1459	0.73	0.1319

Table 5.3: Cross-Validation results for Decision Trees models

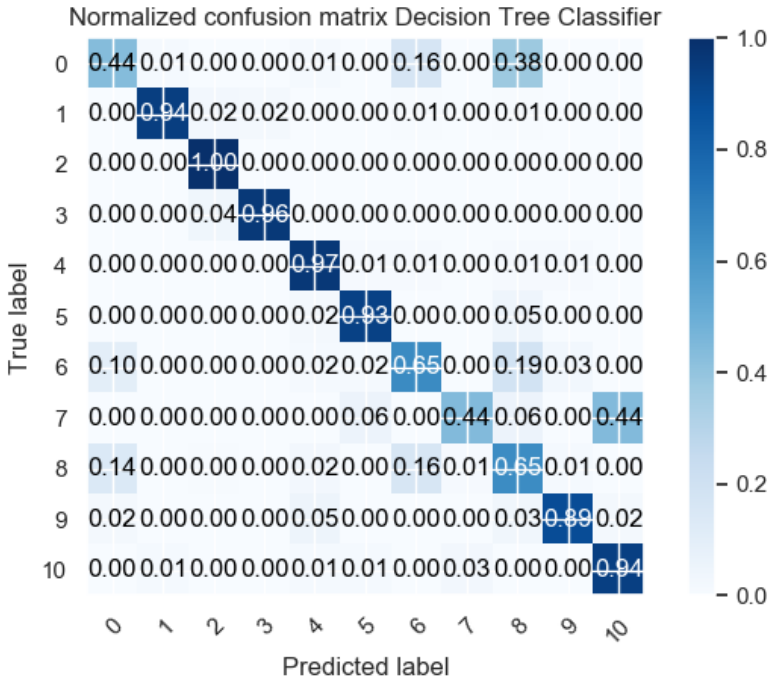

Figure 5.5: Confusion Matrix for Decision Tree using Entropy(C4.5)

- **Random Forest:**

  In this model, it was trained with the algorithms of Gini and Entropy(C4.5) with the same result of Decision trees, and the Entropy was a better precision. So the results mentioned in this document for Random Forest models are using Entropy(C4.5) as the algorithm to measure the quality of a split. The max depth of the trees was tested as 200, 300, and 400, with 300 giving the best results. In table 5.2.5, we can see that the model gave the best result with 200 estimators. Estimators are the number of trees in the forest. In Figure 5.2.5 the confusion matrix shows that only the classes 0 and 7 have a low prediction rate.

No. of estimators	**Velocity and locations**		**Velocity**	
	Precision	Accuracy	Precision	Accuracy
50	0.8097	0.1355	0.7649	0.1315
100	0.8082	0.1332	0.7684	0.133
200	0.8101	0.1436	0.767	0.1337
300	0.8075	0.1458	0.771	0.1339

Table 5.4: Cross-Validation results for Random Forest models

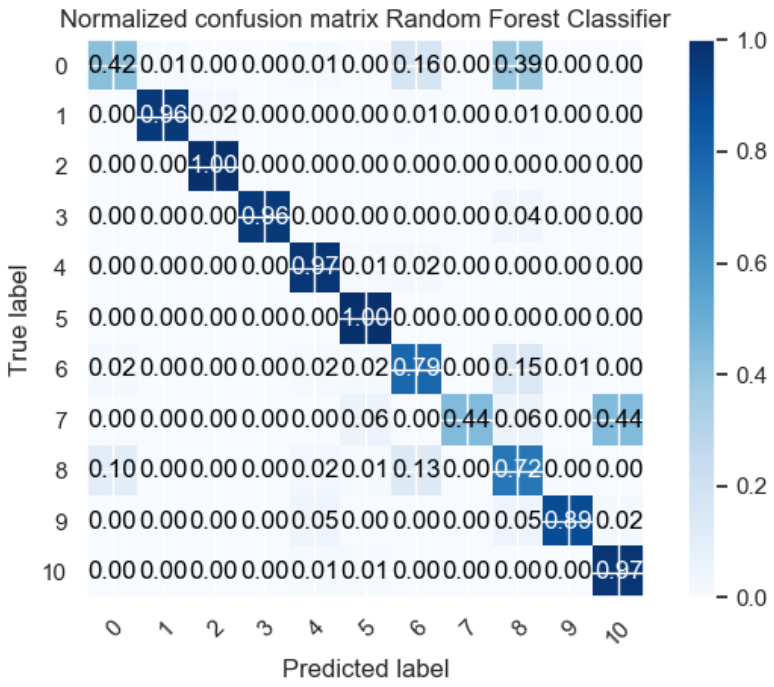

Figure 5.6: Confusion Matrix for Random Forest using 200 estimators

- **Neural Network:**
  For these models, it was trained in various configurations of hidden layers. It was used from 1 to 8 hidden layers with 11, 250, 500, and 750 neurons. Some of the results are in Table 5.2.5. The best result, as we can observe in Table 5.2.5 corresponds to the case with five layers with 500 neurons for which the confusion matrix is presented in Figure 5.7. For some models, it was used the Adam solver and BFGS, but with BFGS having better results. So all results demonstrated are using BFGS solver. As in Decision Trees models, the usage of locations has a good impact on the results of the Neural Network models.

Hidden Layers	**Velocity and locations**		**Velocity**	
	Precision	Accuracy	Precision	Accuracy
None	0.6293	0.147	0.5984	0.1359
11	0.6569	0.1531	0.644	0.1258
4 500	0.7054	0.1479	0.0657	0.1674
5 500	0.7654	0.1535	0.7309	0.1286
6 500	0.7642	0.1421	0.7299	0.1291

Table 5.5: Cross-Validation results for Neural Network models

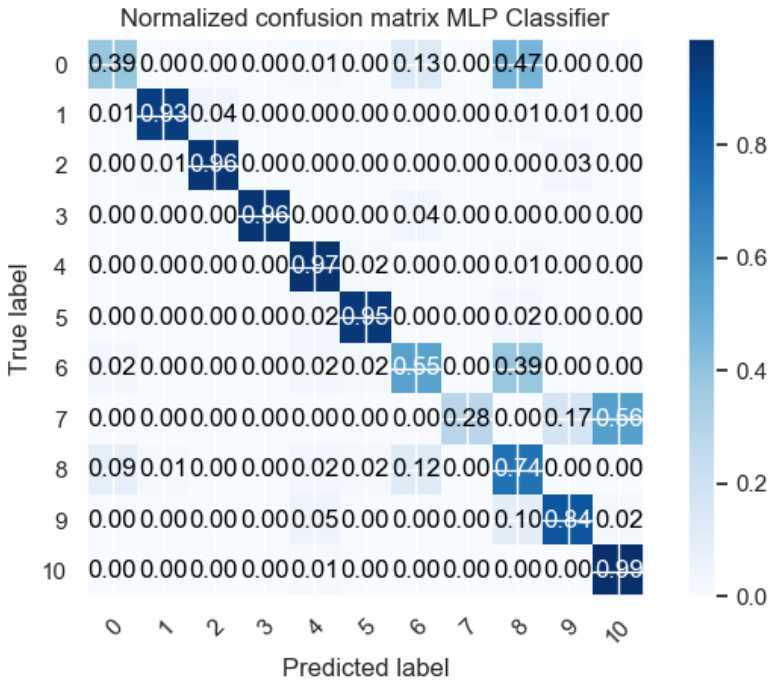

Figure 5.7: Confusion Matrix for Neural Network using 5x500 hidden layers

- **Support Vector Machine:**
  For the built of support vector machine models, it was used the kernel coefficient of Polynomial, RBF, and linear. With the best result as we can observe in Table 5.2.5 is the Polynomial, with its confusion matrix represented in Figure 5.8.

Kernel coefficient	**Velocity and locations**		**Velocity**	
	Precision	Accuracy	Precision	Accuracy
Polynomial	0.702	0.0848	0.6573	0.108
RBF	0.697	0.1092	0.6611	0.12
Linear	0.6018	0.0746	0.5192	0.0263

Table 5.6: Cross-Validation results for Support Vector Machine models

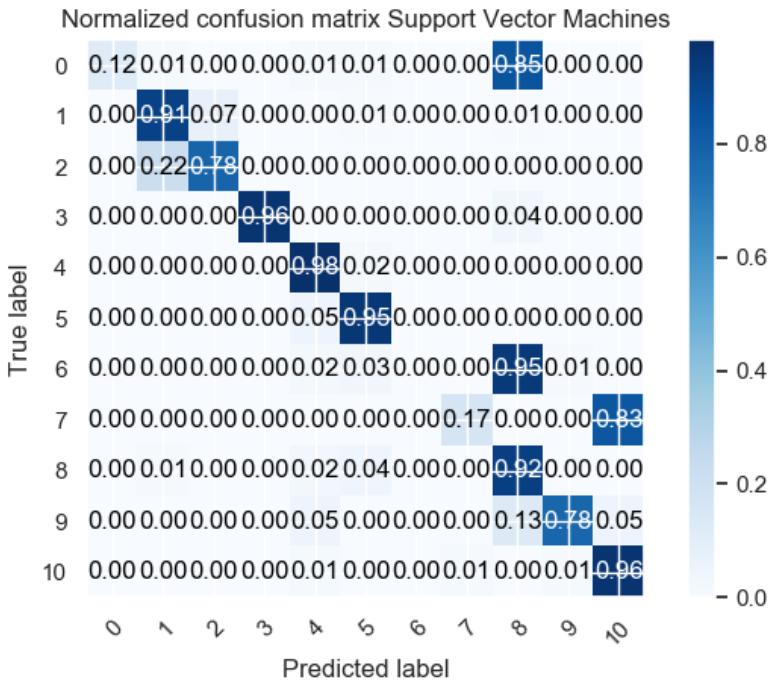

Figure 5.8: Confusion Matrix for Support Vector Machine using Polynomial kernel coefficient

The model with the best result corresponds to the use of a random forest method, with the configuration of 200 estimators. For this model was obtained a precision of 81.01% (accuracy of the evaluation is 0.1436). With this, we can determine that it is possible to create a good model to classify the VMS data as the fishing license.
As seen in Figure 5.6, the classes 0 and 7 have a low true positive rate. The usage of more data and data that are certified that the VMS data of a vessel is only for the propose of the fishing license can improve the precision of the model.

5.2.6 Deployment

To use the trained model, we separated JFA into 4 steps:

1. **Receive VMS data and register in a BD.** JFA needs to be able to receive VMS data so it can classify it, but it is also important to persist the data in a way that we can update the model with more recent data.

2. **Pre process the data.** If SFA is installed in the blue box that is filling the VMS data, the data entries can be already classified as fishing or not fishing. If not, JFA needs to collect data and run its local SFA for each vessel that does not have SFA. If the data is received in one time when the vessel arrives at the port, convert all the data received, as explained in subchapter 5.2.3. If the blue box is broadcasting data when the vessel is on activity, we can classify the data as fishing or not (if SFA not installed in the blue box) and have a subsystem that detects when a vessel fails to send data or have stooped fishing for more than some defined time. This will allow time as the competent authorities will prepare an inspection as marked vessels even before reaching the port.

3. **Use the model to classify.** With the data obtained in the last step, use the model tested chosen in subchapter 5.2.5.

4. **Validate vessel license comparison, send an alert.** Compare the class attributed by the model with the license of the vessel and, if different, register and send a message to the designated authority.

To keep the model up to date, it is essential to train a new model with recent data from time to time. If possible, to know that data is verified by a competent authority, when training the model gives more relevance to this verified data. This is important because the train data that is not verified can be given by vessels that are not respecting there fishing license and this way corrupting the model.

6

Validation

This chapter describes the process used to evaluate the work done in this thesis.

6.1 Validation of Standalone Fishery Analysis

The VMS data are not classified, for the validation and evaluation of SFALib presented in chapter 4, the first step consists of the classification of the data.

The classification consists of 3 classes:

- Class 0 = Fishing in a known area;
- Class 1 = Not Fishing;
- Class 2 = Fishing in a new area.

It was chosen the vessel with id two because it has the most entries in the database provided to this work. In Figure 6.2, the circles represent locations given by the VMS data to vessel two, and the size of the circle, the velocity. This analysis was done using Windows Power Bi [41].

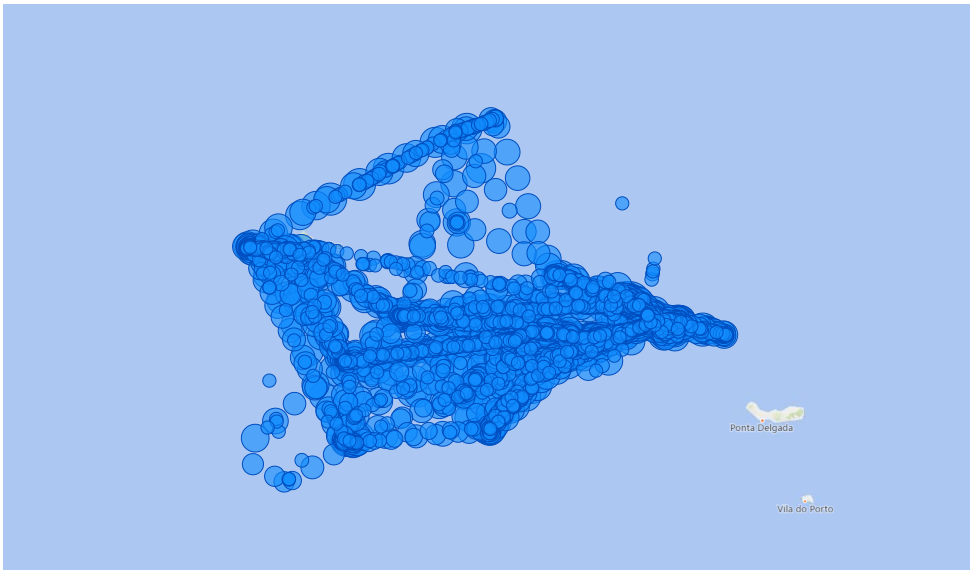

Figure 6.1: Representation of all VMS coordinates for vessel 2

To get data to test SFALib, we need to classify data into three classes.

- Class 0 (Fishing in a known area): To get data to classify as fishing in an area, the data was filtered with a SOG between 0 and 4. Number 4 was chosen because of the analysis made in chapter 3.3, which concludes that the fishing activity done by this vessel is at speeds below four nautical miles. With this, we end up with the data represented in Figure 6.2. To extract the text data, it was chosen the fishing spot near the island of Flores represented in Figure 6.3. 500 random VMS entries were collected at this location without speed restriction. This means that we may have collected travel data, but no speed bias was passed to this test data.
- Class 1 (Not fishing): For with class, it was extracted 500 random VMS entries with the location near Terceira island represented in figure 6.4. It was chosen this spot because most points appear to be on a well-defined trajectory. This pattern contrasts with the random-looking and under-lapping points in Figure 6.3.
- Class 2 (Fishing in a new area): To have data from this class, it was created new data. So 500 new VMS entries were created. The data was created using the data from class 0 but changing to a location near mainland Portugal that is placed with no entries for with vessel.

6.1.1 Validation and evaluation

To evaluate the classification accuracy of SFALib, it was used precision and recall.

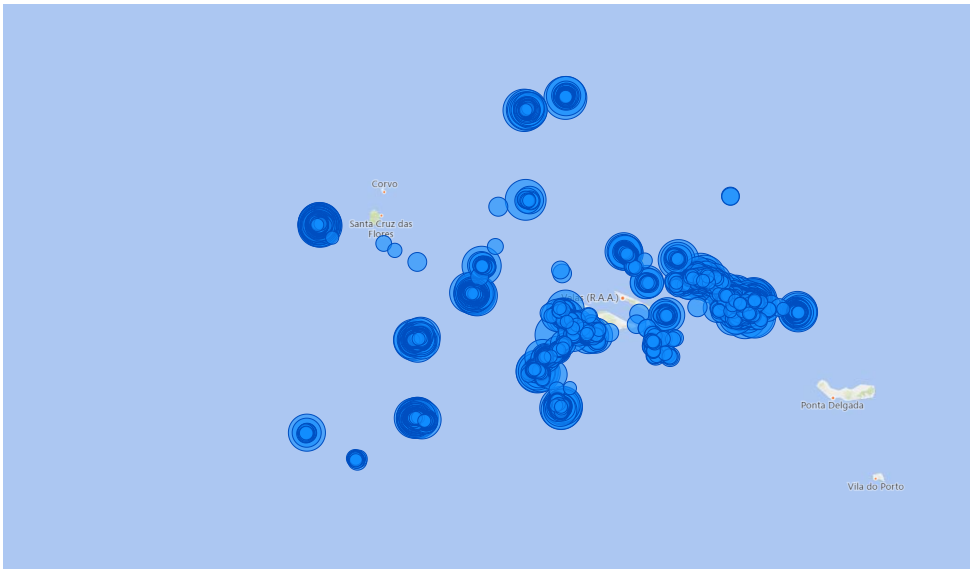

Figure 6.2: Representation of VMS coordinates for vessel 2 with speeds inferior of 4

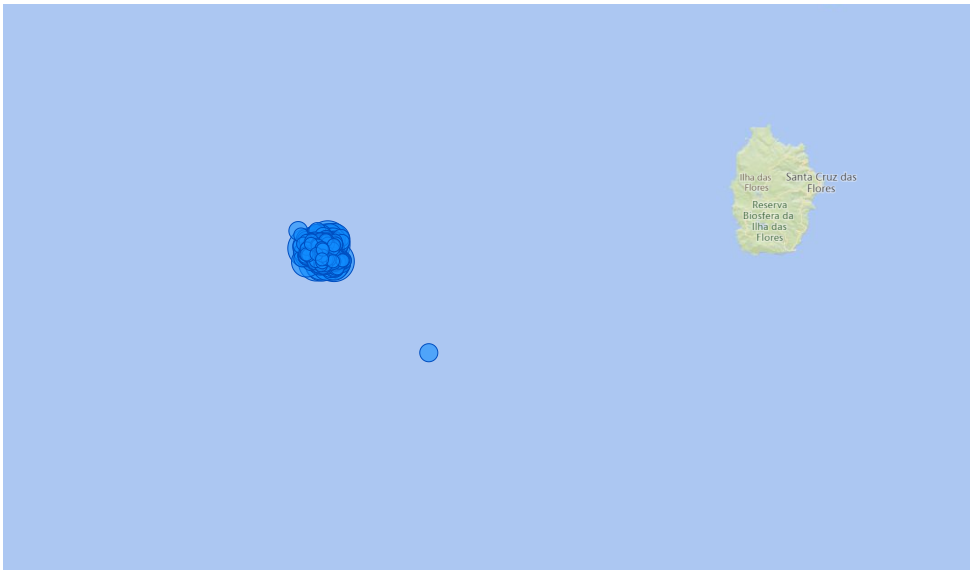

Figure 6.3: Representation of VMS coordinates for vessel 2 with speeds inferior of 4 near Flores island

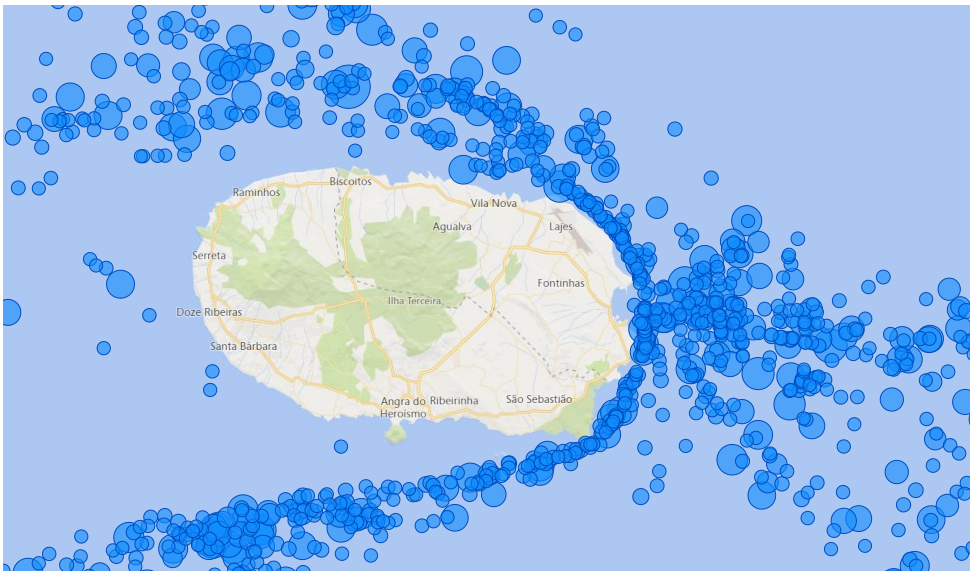

Figure 6.4: Representation of VMS coordinates for vessel 2 near Terceira island

Precision: Proportion of positive identifications that was actually correct is measured as $\frac{tp}{tp+fp}$.
In table 6.1.1 we can see that the lower the configuration value for velocity, the better are the results. This can mean that the gap in speed between fishing and traveling is big.

The best result has an average precision of 0.9.

Configuration for velocity	**Fishing**	**No Fishing**	**Fishing in new area**
0.1	0.67	1	0.67
0.05	0.738	1	0.738
0.01	0.838	1	0.838
0.001	0.85	1	0.85
0	0.85	1	0.85

Table 6.1: Precision per class per configuration

Recall: Proportion of actual positives that was identified correctly is measured as $\frac{tp}{tp+fn}$.
In Table 6.1.1 we can confirm that, like in precision, the recall is better with a low configuration value for velocity.

The best result has an average recall of 0.9231.

Configuration for velocity	**Fishing**	**No Fishing**	**Fishing in new area**
0.1	1	0.6024	1
0.05	1	0.6562	1
0.01	1	0.7553	1
0.001	1	0.7692	1
0	1	0.7692	1

Table 6.2: Recall per class per configuration

In Table 6.1.1, we can see that there is some fishing and fishing classified data that was predicted as not fishing. However, considering that the classified data can have some errors in classification, we cannot assure that the predicted not fishing of fishing classed is not actually, not fishing.

Prediction/Real	**Fishing**	**No Fishing**	**Fishing in new area**
Fishing	0.85	0	0
No Fishing	0.15	1	0.15
Fishing in new area	0	0	0.85

Table 6.3: Confusion matrix for configuration with 0.001 for velocity

6.2 Validation of Joined Fishery Analysis

To validate and evaluation of the model used for Joined Fishery Analysis decided in subsection 5.2.5 as being Random Forest, it was used the same data as in subsection 5.2.5.

In subsection 5.2.5 we use cross-validation [37] that trains the model with approximately the same percentage of samples of each target class as the complete set. In this validation, it was only used one vessel of each license for the train and all of the tests. In this case, we will train the model with 11 vessels but test with 30. This model will have a strong bias to these 11 vessels, but we will see the degradation of the results when testing with these unknown vessels by the model. This is useful to understand the importance of having the model up to date and how badly it preforms with new vessels.

6.2.1 Validation and evaluation

To evaluate the classification accuracy of Joined Fishery Analysis model it was used a confusion matrix and the model precision and recall.

In figure 6.5 we can observe that this model is less accurate that the model tested in Figure 5.6. The precision in this model is 0.6727, and the recall is 0.6379.

This means that different vessels in the same license can operate at different speeds, and so for the model to work without differentiation of vessel power, length, or gross tonnage, it requires that the training dataset is varied in the type of vessels. The other solution is to create different models by categorizing vessels by vessel information(Gross tonnage, length, and power). This way, the models can by specific and achieve better results.

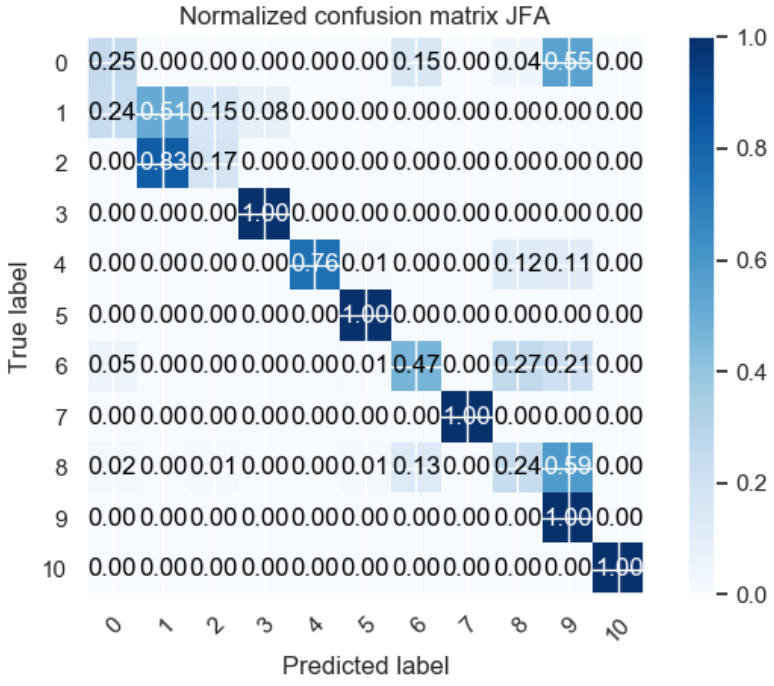

Figure 6.5: Confusion Matrix for Joined Fishery Analysis validation

7

Conclusion

This chapter presents the conclusion of this research, starting with an overview of the work covered in this document and concluding with a set of possible routes for future work.

7.1 Overview

There is great concern about fisheries fraud as demonstrated by the Food and Agriculture Organization of the United Nations [31]. In the European Union through Directorate-General for Maritime Affairs and Fisheries in the European commission, studies are being done and discussing the best way to act on this illegal action using VMS and new technologies like VMS [5].

The objective of this dissertation was to find ways to detect tax fraud in fishing activity efficiently and quickly to be able to detect when a vessel had an operation with abnormal behavior, to be able to notify the authorities while the vessel is still unloading the fish at the dock.

We were able to demonstrate two ways to classify and validate VMS data. The importance of being able to evaluate this type of data will be a great weapon against the tax fraud that occurs in the fishing sector. Another potential return from this work was the possibility to understand the fishing patterns to be able to create plans for environmental protection.

For this work, it was used real VMS data from Portuguese fisheries.

All the objectives proposed in this document have been achieved.

7.1.1 Conclusions of Standalone Fishery Analysis

In SFA, we were able to demonstrate that it is possible to classify in real-time two crucial aspects. If the vessel is fishing and if it is fishing in a new area.
Considering the work done we conclude that classifying if the vessel is fishing at a given moment, taking into account the historical speed of the same, it is possible since we demonstrate that the speed of each vessel has well defined the distribution of fishing speeds, thanks to the fact that the boats spend much of their time fishing. With this information and using clustering algorithms, it is also possible to define fishing areas.

This type of classification is advantageous to understand the fishing patterns in a given area. Another impressive result is the possibility of over the years understand if there are variations in the level of hours of fishing and fishing zones, trying to understand the temporal evolution of the fishing and by consequence of its raw material. With this to understand if the boats spend more time or less inactivity by each time they leave (it can mean that it is becoming easier or more difficult to catch fish) if there is a movement of the activity by type of license (can infer if certain types of fish are disappearing in certain areas and emerging in new areas).

The main weakness of SFA is the lack of classified VMS data. Since it is not possible to test the SFA results with data classified on the ship, it is not possible to measure the real accuracy of the classifier. With classified data, it would also be possible to adjust the classification parameters of the SFA better.

7.1.2 Conclusions of Joined Fishery Analysis

In the second solution, we want to show that it is possible to classify the fishing license by taking into account the VMS data, more precisely speed and position data. The treatment of the data was rewarding since it was possible to find correlations between the type of fishing and the actions of the vessels in fishing activity.

It still takes much work to have variables of enough quality to create a good classifying model. We created different types of data mining algorithms to determine which best fits this problem.

The main weakness of JFA is the prospect of having fraudulent data used in training the model. This could be resolved by training the model with data on which the on-board inspection took place or giving this data more weight in the model than the un-inspected.

7.2 Future Work

In future work, it is imperative to create classified VMS data to analyze the SFA accuracy better. Also, have data from fisheries activity that was inspected by a competent authority to test the models better and if with enough data, to train them with only inspected data or giving more weight to this data so we can have more accurate models.

Bibliography

[1] T. Agardy. Effects of fisheries on marine ecosystems: a conservationist's perspective. ICES Journal of Marine Science, pages 761–765, 2000.

[2] J. R. N. B. S. A. O. R. E. Francois Bastardie. Effects of fishing effort allocation scenarios on energy efficiency and profitability: An individual-based model applied to danish fisheries. In Fisheries Research, volume 106, pages 501–516, September 2010.

[3] Jacob Benesty, Jingdong Chen, Yiteng Huang, and Israel Cohen. Pearson Correlation Coefficient, pages 1–4. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009. ISBN 978-3-642-00296-0. doi: 10.1007/978-3-642-00296-0_5. URL https://doi.org/10.1007/978-3-642-00296-0_5.

[4] Leo Breiman. Random forests. Machine Learning, 45(1):5–32, Oct 2001. ISSN 1573-0565. doi: 10.1023/A:1010933404324. URL https://doi.org/10.1023/A:1010933404324.

[5] European Commission. Control technologies. https://ec.europa.eu/fisheries/cfp/control/technologies_en, 2014. Accessed on 2020-02-01.

[6] Yu-Hong Dai. A perfect example for the bfgs method. Mathematical Programming, 138(1):501–530, Apr 2013. ISSN 1436-4646. doi: 10.1007/s10107-012-0522-2. URL https://doi.org/10.1007/s10107-012-0522-2.

[7] K. S. Alfred M. Duda. A new imperative for improving management of large marine ecosystems. Ocean and Coastal Management, pages 797–833, 2002.

[8] J. e. C. S. Pires. Consumo de peixe em portugal. https://sites.google.com/site/docapescacreative/consumo-de-peixe-em-portugal, 2013. Accessed on 2018-04-06.

[9] ec.europa.eu. European commission. https://ec.europa.eu/fisheries/cfp/control/technologies/vms_en, 2018. Accessed on 2019-01-21.

[10] Theodoros Evgeniou and Massimiliano Pontil. Support vector machines: Theory and applications. In Machine Learning and Its Applications: Advanced Lectures, volume 2049, pages 249–257, 01 2001. doi: 10.1007/3-540-44673-7_12.

[11] Alfredo Alessandrini Fabrizio Natale, Maurizio Gibin. Mapping fishing effort through ais data. PLOS ONE, 2015.

[12] Jinxin Gao and David B. Hitchcock. James–stein shrinkage to improve k-means cluster analysis. Computational Statistics & Data Analysis, 54(9):2113 – 2127, 2010. ISSN 0167-9473. doi: https://doi.org/10.1016/j.csda.2010.03.018. URL http://www.sciencedirect.com/science/article/pii/S0167947310001209.

[13] Matheus Gonzalez. Crisp-dm na prática. https://medium.com/@mgonzaleze/crisp-dm-na-pr%C3%A1tica-65be0ee92ada, 2019. Accessed on 2020-03-05.

[14] Bhumika Gupta, Aditya Rawat, Akshay Jain, Arpit Arora, and Naresh Dhami. Analysis of various decision tree algorithms for classification in data mining. International Journal of Computer Applications, 163(8):15–19, Apr 2017. ISSN 0975-8887. doi: 10.5120/ijca2017913660. URL http://www.ijcaonline.org/archives/volume163/number8/27414-2017913660.

[15] Inmarsat. Inmarsat c. https://www.inmarsat.com/services/safety/inmarsat-c/, 2019. Accessed on 2019-09-11.

[16] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. International Conference on Learning Representations, 12 2014.

[17] Slava Kisilevich, Florian Mansmann, and Daniel A. Keim. P-dbscan: a density based clustering algorithm for exploration and analysis of attractive areas using collections of geo-tagged photos. In COM.Geo, 2010.

[18] Trupti M. Kodinariya and Prashant R. Makwana. Review on determining number of cluster in k-means clustering. In Review on determining number of Cluster in K-Means Clustering, 2013.

[19] David Kriesel. A Brief Introduction to Neural Networks. dkriesel.com, 2007. URL availableathttp://www.dkriesel.com.

[20] Vladimir Kvasnicka, Martin Pelikan, and Jirí Pospíchal. Hill climbing with learning (an abstraction of genetic algorithm). In Hill Climbing with Learning, 1995.

[21] Bin Liu, Ying Yang, Geoffrey I. Webb, and Janice Boughton. A comparative study of bandwidth choice in kernel density estimation for naive bayesian classification. In Thanaruk Theeramunkong, Boonserm Kijsirikul, Nick Cercone, and Tu-Bao Ho, editors, Advances in Knowledge Discovery and Data Mining, pages 302–313, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg. ISBN 978-3-642-01307-2.

[22] M. I. Marzuki, R. Garello, R. Fablet, V. Kerbaol, and P. Gaspar. Fishing gear recognition from vms data to identify illegal fishing activities in indonesia. In OCEANS 2015 - Genova, pages 1–5, May 2015. doi: 10.1109/OCEANS-Genova.2015.7271551.

[23] Microsoft. Sql server. https://www.microsoft.com/pt-pt/sql-server, 2017. Accessed on 2019-08-14.

[24] C. Ulrich N. T. Hintzen, F. Bastardie and N. Deporte. Vmstools: Open-source software for the processing, analysis and visualization of fisheries logbook and vms data. In Fisheries Research, pages 31–43, September 2012.

[25] Raphael Obi Okonkwo and Francis O. Enem. Combating crime and terrorism using data mining techniques. In COMBATING CRIME AND TERRORISM USING DATA MINING TECHNIQUES, 2011.

[26] Oracle. Java 8 central. https://www.oracle.com/technetwork/java/javase/overview/java8-2100321.html, 2018. Accessed on 2019-08-13.

[27] T.R. Patil and Swati Sherekar. Performance analysis of naive bayes and j48 classification algorithm for data classification. Int. J. Comput. Sci. Appl., 6:256–261, 01 2013.

[28] G. J. Piet and F. J. Quirijns. The importance of scale for fishing impact estimations. Canadian Journal of Fisheries and Aquatic Sciences, 66(5):829–835, 2009. doi: 10.1139/F09-042. URL https://doi.org/10.1139/F09-042.

[29] C. Pimenta. Esboço de Quantificação da Fraude em Portugal, pages 1–44. Edições Húmus, 2009.

[30] PostgreSQL. Postgresql. https://www.postgresql.org/, 2019. Accessed on 2019-08-14.

[31] Alan Reilly. Overview of food fraud in the fisheries sector. In FAO Fisheries and Aquaculture Circular No. 1165, December 2018.

[32] L. Rokach and O. Maimon. Ieee transactions on systems, man, and cybernetics—part c: Applications and reviews publication information. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 35(4):c2–c2, Nov 2005. ISSN 1094-6977. doi: 10.1109/TSMCC.2005.859799.

[33] Lior Rokach and Oded Maimon. Data Mining With Decision Trees: Theory and Applications. World Scientific Publishing Co., Inc., River Edge, NJ, USA, 2nd edition, 2014. ISBN 9789814590075, 981459007X.

[34] Tommaso Russo, Lorenzo D'Andrea, Antonio Parisi, and Stefano Cataudella. Vmsbase: An r-package for vms and logbook data management and analysis in fisheries ecology. PLOS ONE, 9(6):1–18, 06 2014. doi: 10.1371/journal.pone.0100195. URL https://doi.org/10.1371/journal.pone.0100195.

[35] John G. Saw, Mark C. K. Yang, and Tse Chin Mo. Chebyshev inequality with estimated mean and variance. The American Statistician, 38(2):130–132, 1984. ISSN 00031305. URL http://www.jstor.org/stable/2683249.

[36] N. S. A. J. T. P. S. S.-P. Yashashwita Shukla. Big data analytics based approach to tax evasion. International Journal of Engineering Research in Computer Science and Engineering, pages 56–59, 2019.

[37] M. Stone. Cross-validatory choice and assessment of statistical predictions. Journal of the Royal Statistical Society. Series B (Methodological), 36(2):111–147, 1974. ISSN 00359246. URL http://www.jstor.org/stable/2984809.

[38] Monika Jena Swasti Singhal. A study on weka tool for data preprocessing, classification and clustering. International Journal of Innovative Technology and Exploring Engineering (IJITEE), 2(6):250–253, 2013. ISSN 2278-3075. URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.687.799&rep=rep1&type=pdf.

[39] Prajwala Talanki. Comparative analysis of em clustering algorithm and density based clustering algorithm using weka tool. International Journal of Engineering Research and Development, 9:2278–800, 02 2014.

[40] Waikato University. Weka 3. https://www.cs.waikato.ac.nz/ml/weka/, 2019. Accessed on 2019-08-14.

[41] Windows. Power bi. https://powerbi.microsoft.com/pt-pt/, 2019. Accessed on 2019-10-06.

[42] R Wirth and Jochen Hipp. Crisp-dm: Towards a standard process model for data mining. Proceedings of the 4th International Conference on the Practical Applications of Knowledge Discovery and Data Mining, 01 2000.

[43] Xsealence. Xsealence. http://www.xsealence.pt/portfolio/monicap/, 2018. Accessed on 2019-07-31.

[44] Xsealence. Xsealence. http://www.xsealence.pt, 2018. Accessed on 2019-09-11.

[45] Michael Ringgaard Yin Wen Chang, Cho Jui Hsieh. Training and testing lowdegree polynomial data mappings via linear svm. Journal of Machine Learning Research, 11:1471-1490, 2010. URL http://www.jmlr.org/papers/volume11/chang10a/chang10a.pdf.

Appendix A

A.1 Discrimination of fishing licenses

Siege: The purse seine used on the mainland is characterized by the use of a catch at the bottom of the net - this allows the net to be closed like a bag in order to retain the catch.

Dragging: **Drag of Doors:** A bottom-trawl net towed by a single vessel, the horizontal opening of which is ensured by relatively heavy trawl doors, which may be fitted with a steel shoe designed to withstand a contact with the bottom. **Pole drag:** Rod trawling is characterized as a medium-sized trawl art where the mouth, devoid of wings, is held open by the action of two rods or a horizontal rod and rigid lateral structures. Dredge: Small and medium-sized trawling art in which the mouth is composed of a rigid structure and the bag is mesh or made up of a metal grid.

Gillnets and Trammel nets: Fishing method using a rectangular net with one, two or three rafts held upright by floatation cables and cables of used ballast insulated or in hunting.

Fishhook: A fishing method that uses lines and, in general, one or more hooks, ballasts and buoys. It can be practiced with gear that is integrated in the following groups: troll, cane and hand line, longline, tone and fishing nipple.

Traps: **Cage Traps:** Fishing method by which the prey is attracted or referred to a device that prevents leakage. **Shelter Traps:** Fishing method by which the prey is

attracted or referred to a device, in this case the pots.

Sliding Enclosures: Método de pesca que utiliza uma estrutura de rede com bolsa e grandes asas laterais que arrastam e, previamente ou em simultâneo, envolvem ou cercam.

Catch: Uses several simple utensils. It can be practiced by an individual, using or not a support vessel and apnea diving equipment.

This data is available in https://www.dgrm.mm.gov.pt

Appendix B

The purpose of this appendix is to explain in more detail how the algorithm works used in SFALib described in sub chapter 4.3.

1. The algorithm starts by creating an histogram with the vessel velocity data as the one in Figure B.1.

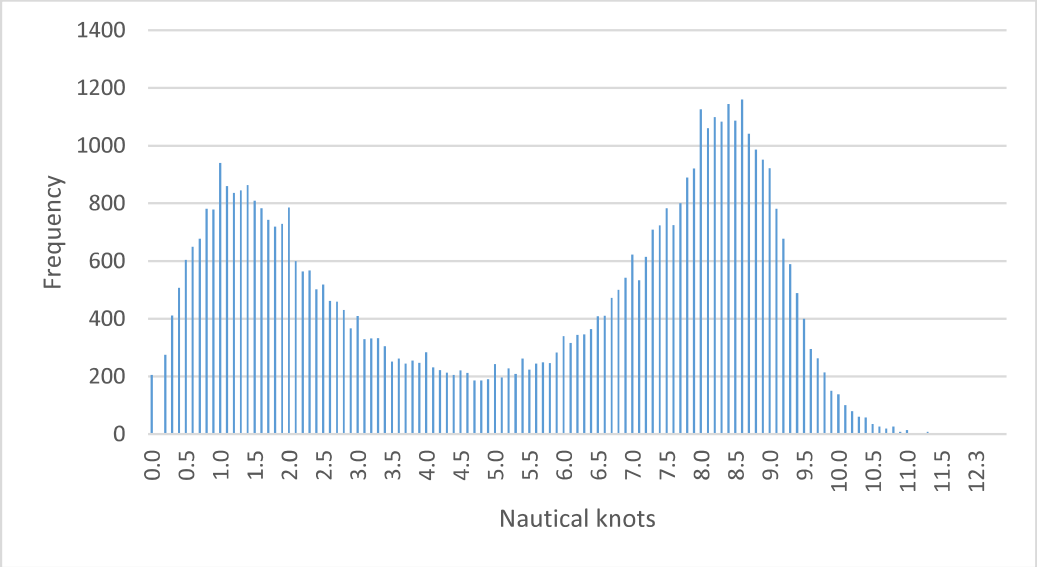

Figure B.1: Create histogram

2. Then remove from the histogram the frequency of value 0.

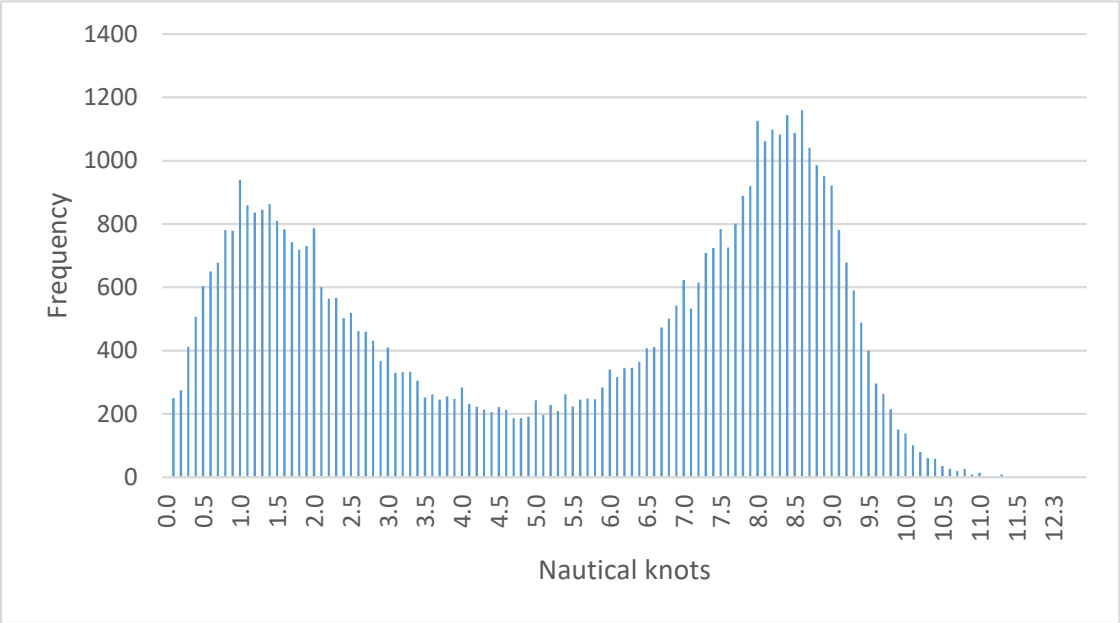

Figure B.2: Remove requency of value 0

3. Use Hill-Climbing function to get maximum frequency for the velocity of fishing activity.

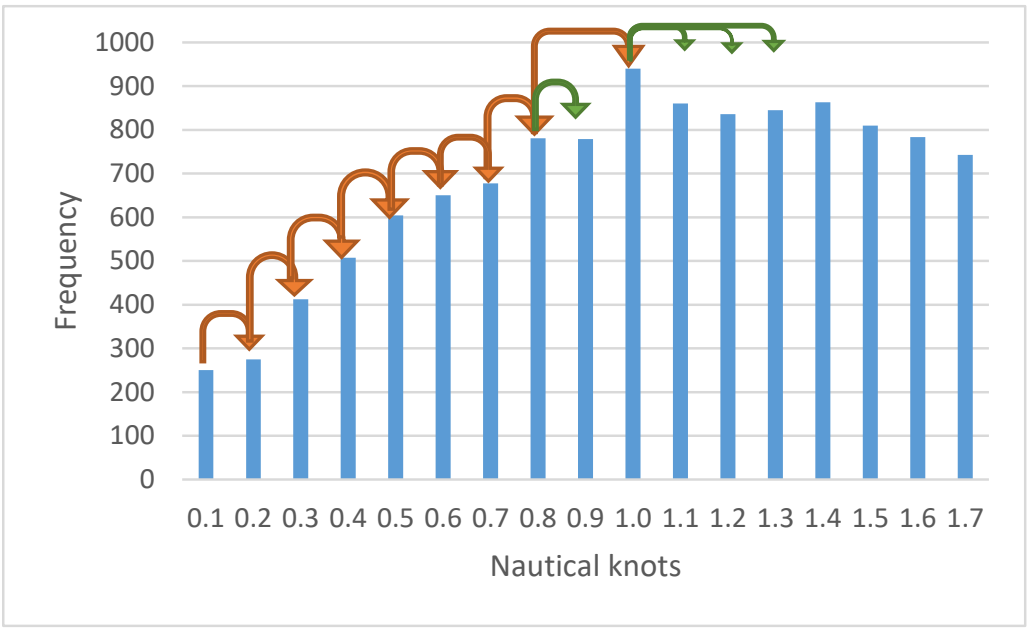

Figure B.3: Find maximum

4. After finding the maximum frequency Hill-Climbing find the next minimum frequency to represent the maximum velocity value of the fishing activity.

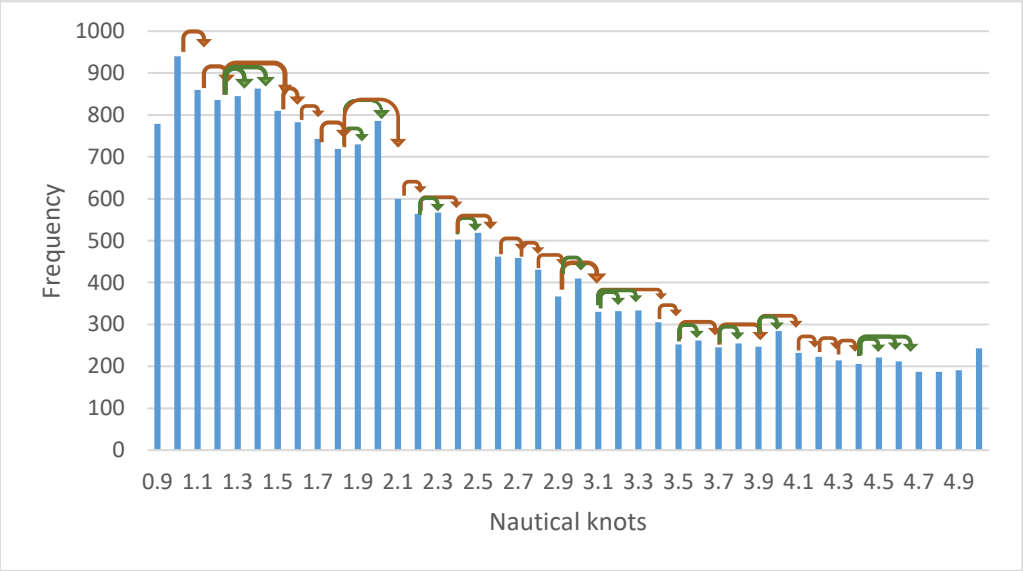

Figure B.4: Find minimum

5. Remove values greater that the maximum velocity value found in the last step.

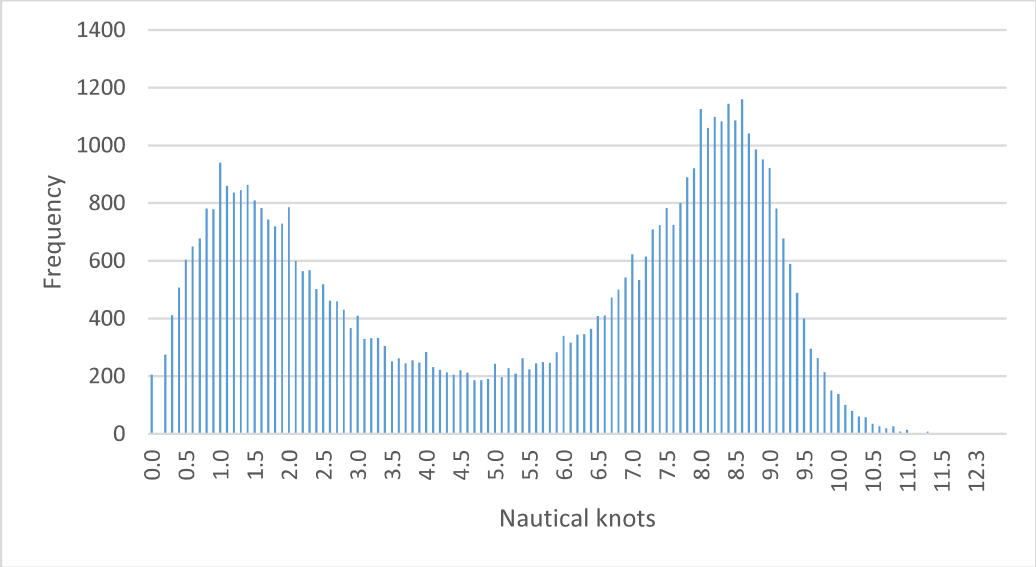

Figure B.5: Remove verge values

6. Calculate Kernel Density distribution.

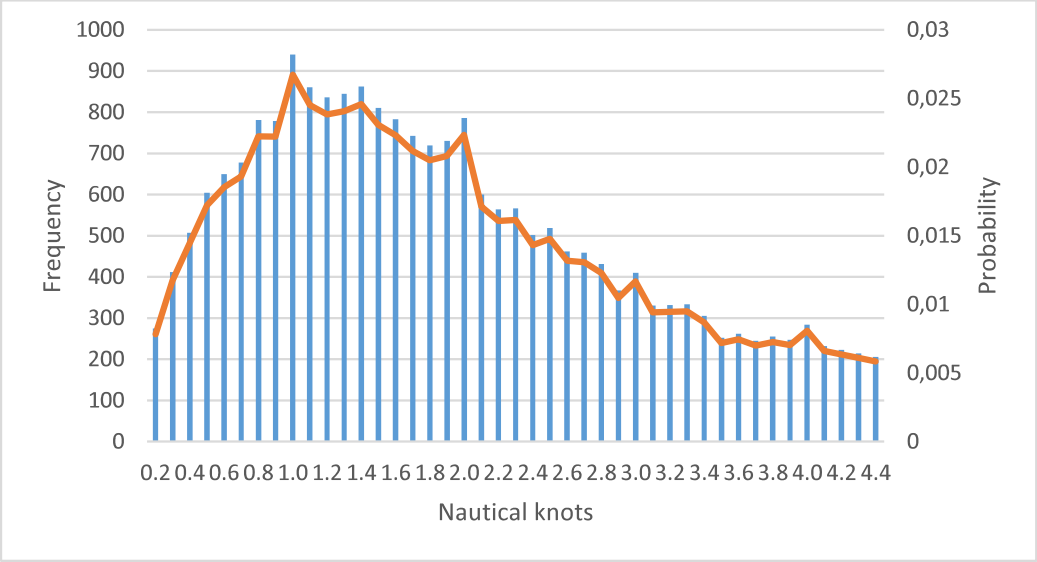

Figure B.6: Kernel Density distribution

7. Calculate cumulative function of Kernel density distribution.

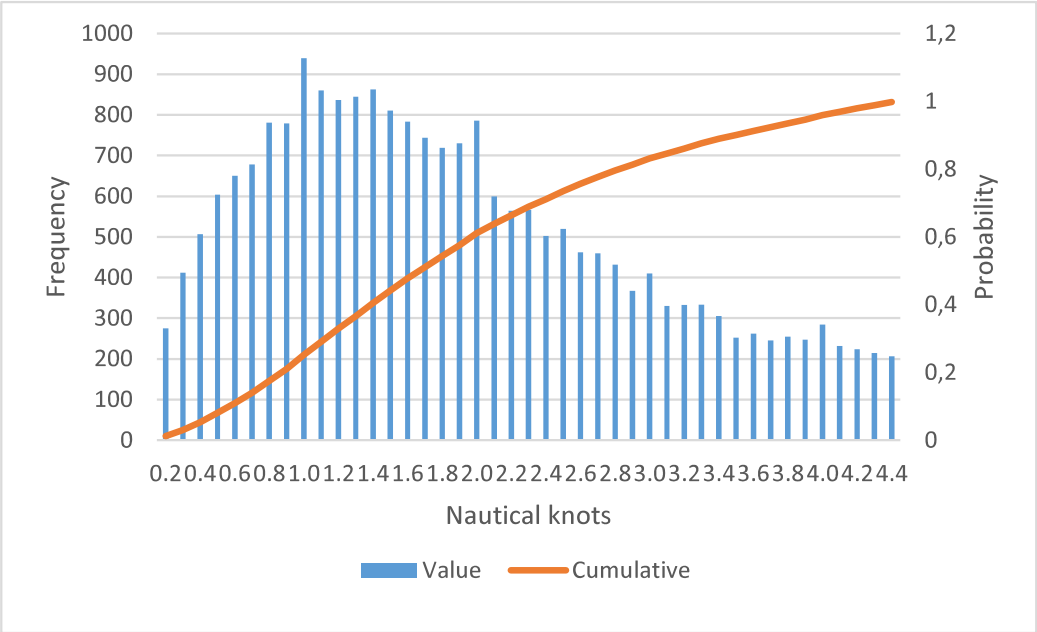

Figure B.7: Cumulative Kernel Density

8. Set margin (configurable, exemple 20%) of the cumulative function and get fishing velocity limits of the vessel.

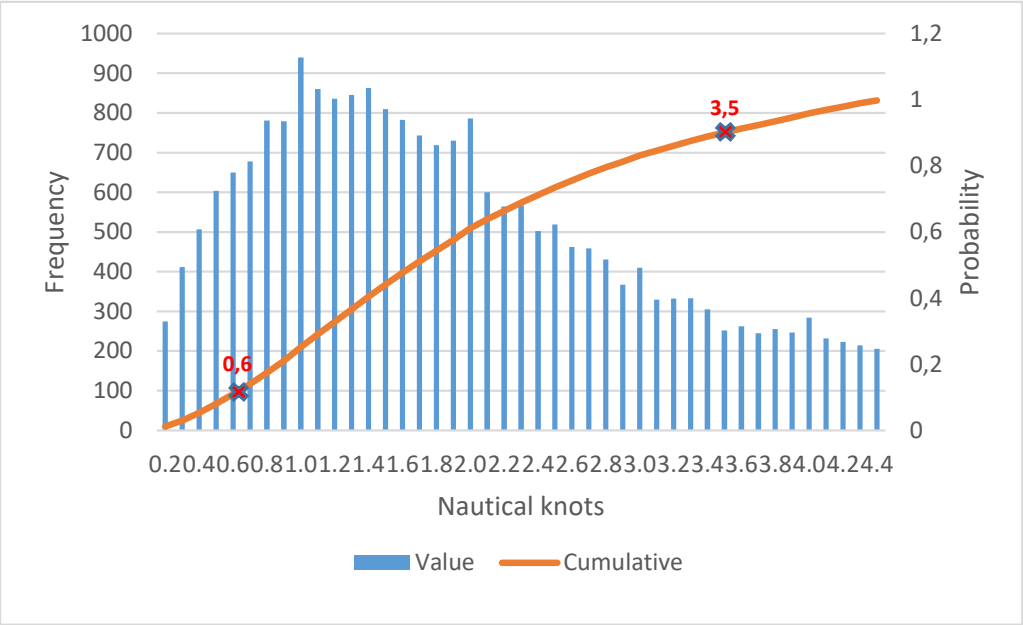

Figure B.8: Set margin and get limits