PREDICTING CATCH-PER-UNIT-EFFORT USING SEMANTIC TRAJECTORIES AND MACHINE LEARNING

by

Pedram Adibi

Submitted in partial fulfillment of the requirements
for the degree of Master of Computer Science

at

Dalhousie University
Halifax, Nova Scotia
April 2020

© Copyright by Pedram Adibi, 2020

Table of Contents

List of Tables

List of Figures

Abstract

In this study, we present a framework for the prediction of catch-per-unit-effort (CPUE)—an important index in the assessment of fisheries resource exploitation—using three data sources from the North Adriatic region: fishing-vessel tracking data (obtained from terrestrial Automatic Identification System (AIS)), the associated daily landing reports (i.e., the amount and species of fish caught by each vessel), and the relevant environmental data. As a part of this framework, two high-level spatio-temporal representations of the data were constructed through the use of semantic trajectory modelling and fusion of the data sources—namely a set of enriched semantic trajectories of the fishing trips, and gridded spatio-temporal maps of CPUE. While both representations can have various applications in fisheries management, here they were used for the task of CPUE forecasting, which is the objective of this study. Our prediction results demonstrate the potential of Machine Leaning methods for this task. However, we consider the results to be preliminary due to the limited two-year temporal horizon of the available data, and also with respect to the broader set of possible forecasting techniques. To address these limitations, we also propose several approaches to be employed in the future to expand and improve this work, some of which will be particularly useful with the availability of more data. Similar data could also be available for other regions with intense fishing activity; and the fisheries management in such areas could use methods similar to the ones used in our framework to facilitate data-driven and evidence-based policy making.

List of Abbreviations Used

AIS Automatic Identification System

chl-a Chlorophyll-a

CPUE Cath Per Unit Effort

GPS Global Positioning System

sst Sea surface temperature

Acknowledgements

I would like to thank NSERC (Natural Sciences and Engineering Research Council of Canada) for their financial support.

My deepest gratitude goes to my supervisor, Dr. Stan Matwin, whose commitment to the success of his students goes beyond academic guidance. I am grateful for his tremendous support through a particularly difficult time in my life and helping me to not lose sight of my goal.

Finally, I would like to express my appreciation for the support and encouragement I received from my partner and parents. This accomplishment would have not been possible without them.

Chapter 1

Introduction

The economic significance of fisheries as an exhaustible resource underscores the importance of fisheries management to ensure a sustainable exploitation of the marine resources. Often used in fisheries management, Catch-Per-Unit-Effort (CPUE) is an important index in the assessment of fisheries resource exploitation [16]. CPUE represents the amount of catch relative to the intensity of the effort exerted in the fishing activity leading to the catch. Intuitively, a decline in CPUE implies over-exploitation of the fishery resources; a steady CPUE indicates a sustainable fishing operation; and an increase in CPUE suggests a growing fish population. By quantifying the pressure of fishing activities on the marine resources, CPUE allows for the assessment of the sustainability of fishing operations. Therefore, accurate forecasting of CPUE plays an important role in fisheries management by guiding the policy makers in developing polices that ensure a sustainable fishing industry based on the forecast outcome. This study provides a framework for CPUE prediction in the North Adriatic sea through the integration of relevant data sources, vessel trajectory modeling, and the use of Machine Leaning methods.

The Adriatic sea—the northernmost arm of the Mediterranean sea—accounts for 14% of the fishing fleet that operate in the Mediterranean and Black Sea [29]. Of the total estimated value of US$3.09 billion in fish landings across the two seas in 2016, US$979 million (32%) was attributed to the Adriatic sea [29]. This highly productive area is known to be among the most over-exploited marine resources in the Mediterranean sea. Human activities, intensive fishing, and habitat degradation have contributed to the steady decline in fish populations in the Adriatic sea, and the increasing stress on this vital marine resource [17, 65, 66, 76].

The historical mismanagement of the fishing industry in the region is a result of complex sociopolitical factors which are beyond the scope of this study.[1] However,

[1]Paper [12] provides a context regarding the fisheries management issues in the Adriatic sea.

in recent years there have been developments in regional policies and management strategies towards achieving a more sustainable fishing industry in the Adriatic sea.[2] Most of these policies aim to protect the juvenile and undersized specimens of the commercial species to ensure a stable stock replacement rate [66]. Measures such as mesh size regulations, and instituting minimum catch size requirements, are intended to discourage the catch of juvenile specimens. Furthermore, other measures attempt to protect the nurseries of commercial species through limiting fishing activities. These measures include closure of *areas* (e.g. permanent ban on trawling within 3 nautical miles of the coast, closure of juvenile congregation areas), and *seasons* (e.g. ban on towed gear during part of the summer) [7]. Designing such policies—that attempt to ensure stable stock replacement rates—must be informed by the current and the projected state of stock abundance. In this context, forecasting CPUE is of utmost relevance; since it is considered to be the main index in the assessment of stock abundance [16].

The objective of this project, prediction modeling of CPUE, is motivated by an especially valuable dataset which was obtained by the fusion of three data sources: fishing-vessel tracking data, the corresponding landing reports (i.e.,species and the quantity of fish catches), and the related environmental data. Vessel tracking technologies, such as Automatic Identification System (AIS) [5], have become a primary source of data for scientific works involving fishing activities. In this study, we were provided with *terrestrial* AIS data; i.e., AIS data emitted by the vessels that was received by ground stations located on the Italian coast of the North Adriatic sea. Terrestrial AIS datasets are considered to be high quality mobility data sources because of their high temporal resolution (having update frequency from 2 seconds to two minutes). In contrast, data from satellite AIS (S-AIS) and other VMS (Vessel Monitoring System) technologies usually have much lower temporal resolution (minutes to hours). In this project, the AIS data is used to reconstruct the vessels' fishing trajectories in time and space. Another important source of data in this study is the landing reports dataset from the fish market at Chioggia port, which is the main fishing port on the Italian coast of North Adriatic sea. The AIS and landing reports datasets are integrated to create a high-resolution spatio-temporal CPUE dataset.

[2]The entrance of Croatia into the EU in 2013 facilitated easier agreement between Italy and Croatia—the two main players in the Adriatic sea, accounting for 99% of its fishing activity [12, 29].

The generated spatio-temporal dataset is further enriched with environmental factors such as daily sea surface temperature, chlorophyll-a concentration, and wave height. The resulting enriched spatio-temporal dataset is then used for prediction modeling of CPUE.

We consider our prediction results to be preliminary due to the short temporal span of the available data (two years: 2015-2016); which in turn limits the set of suitable techniques for the prediction task. However, similar data could be available for other regions with intense fishing activity; or could become available as more local authorities implement regulations requiring the monitoring of fishing activities. For instance, the availability of the AIS dataset used in this project is due to a 2014 European Union regulation which requires fishing vessels of 15 meters or larger to carry an AIS device [28]. Almost all commercial fishing vessels in the area are subject to this strict regional regulation, in comparison to the much more tolerant IMO (International Maritime Organization) regulation that only requires ships of 300 or more gross tonnage and passenger ships to carry AIS [41]. This study demonstrates the potential of mobility data analysis and machine learning to provide fisheries management with valuable insight, and facilitate evidence-based policy making. Considering the significant environmental and economical effects of such policies on a micro and macro scale, data-driven and evidence-based approaches in fisheries policy making are extremely important.

In the light of the three valuable data sources previously mentioned, the two following main research questions will shape this study.

1. What kinds of information can be extracted from the combination of the available data sources to assist policy makers by improving our understanding of the spatio-temporal aspects of fishing activities in the North Adriatic sea?
2. How can we perform CPUE prediction using the available data?

Data-driven answers to these question can be sought through the application of mobility data analysis, data integration, and Machine Learning methods. This work mainly addressed the second question by focusing on trajectory modeling of fishing vessels, and machine learning techniques for prediction modeling of CPUE. The first question is briefly addressed in the Future Work (Chapter 6), where we discuss other potential applications of this rich blend of data sources in assisting with fisheries

policy making.

In this work, we propose a framework for predicting CPUE using the available data sources, contributions of which are as follows:

1. semantic trajectory modeling of fishing vessels to extract mobility knowledge relating to fishing activities;
2. integration of heterogeneous data sources to provide more insightful information for fisheries management;
3. data modeling conducive to the use of Machine Learning for the spatio-temporal prediction of CPUE;
4. applying Machine Learning techniques to perform CPUE prediction by learning from the integrated dataset;
5. publication of a paper in the International Workshop on Multiple-Aspect Analysis of Semantic Trajectories [1].[3]

A bird's eye view of the framework is given in Figure 1.1. The figure is divided in three sections. The top section demonstrates the three data sources that were used in this study, i.e., (i) the vessel tracking (AIS) dataset, (ii) the landing reports dataset from the Chioggia fish market, and (iii) the relevant environmental variables. The middle section of Figure 1.1 lays out the process of semantic trajectory modeling and data fusion. The trajectories are constructed from the AIS dataset, and enriched with semantic information such as vessel activity (e.g., fishing, navigating, etc.), and the amount of fish caught on each fishing segment of the trajectories. Using the resulting semantic trajectories and a spatial grid, spatio-temporal maps of CPUE are constructed, and then augmented with the environmental variables. The bottom section of Figure 1.1 shows the predictive modelling step using the resulting spatio-temporal dataset and machine learning methods. The prediction models are trained on data from 2015 (the first of two years with available data), and evaluated on 2016.

This document is structured as follows. Chapter 2 provides the background and the related work regarding CPUE forecast and trajectory modelling. Chapter 3 describes the raw datasets, and the process of obtaining the high-resolution spatio-temporal CPUE datasets from the raw data (corresponding to the top and middle sections of Figure 1.1). Chapter 4 states the prediction problem, and how regression

[3]This thesis presents our published work ([1]) in more detail and with additional experiments.

modelling is used to perform the prediction task. Chapter 5 describes the machine learning methods employed, and provides a discussion on the forecasting results. Chapters 4 and 5 correspond to the bottom section of Figure 1.1. Finally, Chapter 6 briefly reiterates what was done, discusses the limitations of the study, and proposes ideas for future work.

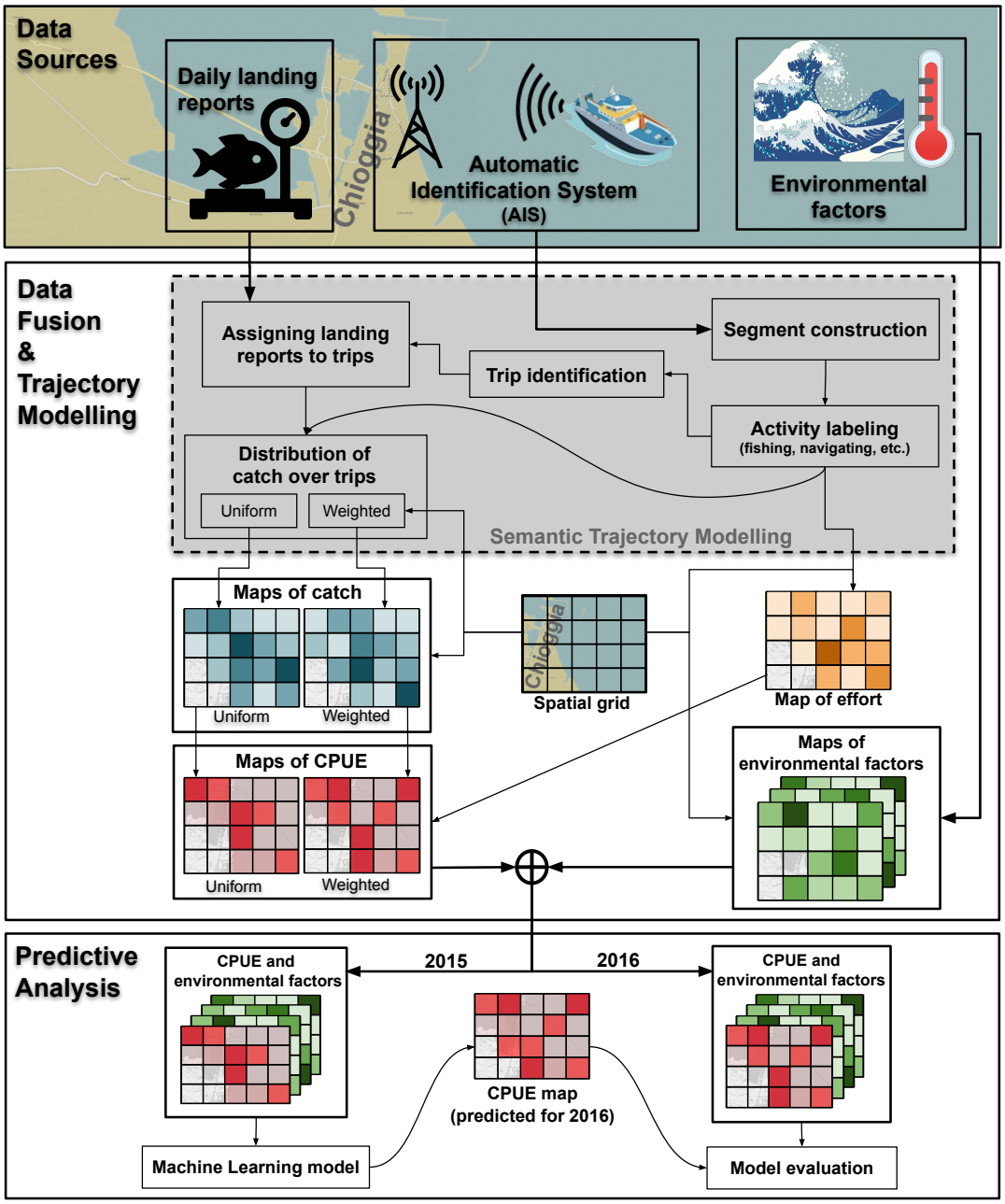

Figure 1.1. Overview of the framework for predicting CPUE

Chapter 2

Background and Related Work

In this chapter, we discuss the background and related work regarding two concepts important for the analysis of fishing activities: (i) Catch-per-unit-effort (CPUE) forecasting, which is the objective of this project; and (ii) fishing vessels trajectory modeling, which we employ in the construction of our CPUE datasets.

2.1 Forecasting CPUE

Catch-per-unit-effort (CPUE) is an indicator of the species abundance in the assessment of fishery resources; and can give information about the sustainability of the fishing activities in the geographical area of interest [6, 26, 22]. In fisheries science, CPUE is often calculated as the ratio of *catch* to *effort*; where *catch* is the amount of fish caught and, *effort* is a measure that quantifies the effort exerted on the fishing activity leading to the catch. Quantification of fishing effort is done differently for various types of fishing equipment, resulting in different types of CPUE. For example, CPUE for trawler and long-liners fishing are often calculated respectively as catch-per-kilometer-towed or catch-per-hook [26].

CPUE is often used as in index to evaluate population trends; where a decrease in CPUE would suggest over-exploitation, a steady CPUE value would suggest sustainable exploitation, and an increase of CPUE would suggest growing population [77]. Therefore, accurate forecast of CPUE can help policy makers maintain a sustainable fishing industry by adapting the fisheries management plans accordingly.

2.1.1 Time-series vs. spatio-temporal CPUE data

CPUE data often consists of values calculated over regular time intervals for the whole geographical area of interest. Such data can be represented as a univariate time-series—i.e., a sequence of values in time. Naturally, many CPUE forecasting approaches use time-series analysis methods such as ARIMA (AutoRegressive Integrated

Moving Average). Works that employ such techniques include [74, 72, 4, 58, 59]. Paper [20] compares linear (ARIMA) and non-linear (ANN) univariate time-series methods for short-term forecasting of halibut CPUE. CPUE datasets used in the aforementioned studies have low temporal resolution (monthly, seasonal, or annual), and CPUE values correspond to the whole geographic study area—that is, no spatial dimension. In contrast, data sources used in this study make it possible to not only calculate CPUE for an arbitrarily small temporal resolution, but also with our choice of spatial granularity. The resulting CPUE dataset is spatio-temporal, for which the standard time-series forecasting methods cannot be used.

Spatio-temporal CPUE datasets are very uncommon in the literature. The few studies we found that utilize spatio-temporal CPUE data, do so by combining independent time-series CPUE data obtained from different areas. For example, [35] uses time-series CPUE data reported by several countries with fishing operations in the Atlantic ocean, and compiles a spatio-temporal dataset by placing the data into coarse time-area strata—monthly time granularity, and 5° latitude–longitude spatial cells (approximately 560 km). Another study, combines time-series CPUE data from three different sites to study the spatio-temporal variability of Atlantic cod [63]. Paper [73] uses spatio-temporal CPUE that is calculated monthly for 1° (approximately 111 km) square grid cells. The spatio-temporal CPUE data used in all the aforementioned studies have very coarse temporal and spatial resolution. Besides, the goal of none of the studies was forecasting. In fact, to the best of our knowledge, no work has been done on high-resolution spatio-temporal forecasting of CPUE.

The scarcity of high-resolution spatio-temporal CPUE datasets is because creating such dataset requires information that is rarely available. More specifically, we would need to have high resolution vessel tracking data (GPS coordinates), along with localized catch amounts at all the points on the vessel trajectories. Most datasets from commercial fisheries either do not include geographical coordinates, or use coarse-scale grids to log fishing activities [51]. However, some highly regulated regions, such as the European Union, have adopted policies that enforce carrying tracking devices for most fishing vessels [28]. Detailed tracking data coupled with vessel's corresponding catch information would make it possible to construct high-resolution spatio-temporal CPUE datasets. Indeed, as will be described in Chapter 3, one focus of this study is

to create high-resolution spatio-temporal CPUE maps using vessel tracking and catch datasets.

2.1.2 Use of environmental factors

Food source and water temperature are understood to be important factors in fish growth and abundance [39], which in turn affect catch rates. At the base of the aquatic food chain are phytoplankton; which are considered to be the *primary producers*, feeding the majority of aquatic life [46]. Similar to plants, phytoplankton convert sunlight to chemical energy using light-harvesting pigments, primarily *chlorophyll-a (chl-a)*. Therefore, chl-a concentration in water can be used as an index for phytoplankton biomass. Effects of the two oceanographic parameters, Sea Surface Temperature (sst) and chl-a concentration, are commonly considered in marine fisheries research. For example, [43] found strong association between weekly sst, chl-a concentration, and daily catch of some small pelagic fish species in the gulf of California. In another study, sst and chl-a concentration were used to forecast fishing grounds off of Gujarat coast in India, potentially increasing the catch amounts by 2-3 folds [69].

On the other hand, weather conditions influence fishing vessels activities, therefore affecting catch rates. Most importantly, turbulent water adversely affect vessels safety, deterring fishing activity when waves are dangerously high. In fact, wave height is found to be inversely related to catch rates [67]. *Significant wave height* is a statistical quantity which is commonly used as a measure of the ocean waves height. This measure is defined as mean of the highest third of wave heights (trough to crest) that occur in a given time period.

Significant wave height can be measured directly by satellite radar altimeters. sst and chl-a concentration can also be measured using satellites equipped with the proper optical sensors. In our approach to forecast CPUE, we use sst, chl-a concentration, and significant wave height data obtained from satellite imagery.

2.2 Fishing vessels trajectory modeling

Thanks to the recent abundance and ubiquity of devices with integrated Global Positioning System (GPS), a wealth of mobility data is being generated for a variety of moving objects. The literature on the broad subject of mobility data mining and

trajectory analysis is vast; only a small subset of which is relevant to our application. This section provides the necessary background and an overview of the literature related to trajectory modeling with a focus on fishing vessel trajectories. More specifically, our focus is on trajectory models that can facilitate creating a high-resolution CPUE dataset by incorporating the information needed for the task, such as vessel fishing activity and catch amounts.

2.2.1 Automatic Identification System (AIS)

When it comes to vessels trajectories, a notable data source is information collected from ships equipped with Automatic Identification System (AIS) [5]. An AIS device is essentially a GPS tracking device that is integrated with other navigation sensors (e.g. gyrocompass), and a VHF (Very High Frequency) transceiver. AIS devices use a standardized, open, and un-encrypted protocol to broadcast the carrying vessel's information, and receive information from other AIS-equipped vessels' in their range. The un-encrypted protocol of AIS sets it apart form other proprietary VMS (Vessel Monitoring System) technologies, and makes it an accessible data source for academic studies.

Broadcast information of an AIS device include a unique vessel identifier number called MMSI (Maritime Mobile Service Identity), geographical coordinates, timestamp, and course information (such as true heading, rate of turn, speed, etc.). Broadcast update rate varies from 2 seconds when vessel is moving fast or maneuvering, to 3 minutes when anchored or moored; giving AIS data a high temporal resolution compared to other VMS technologies, which have update rates of minutes to hours [42]. AIS was initially developed as a safety system for vessel collision avoidance by allowing vessels to detect each other; supplementing radar which has a shorter range. International Maritime Organization (IMO) requires AIS on ships with 300 or more gross tonnage (GT), and all passenger ships [41].

In addition to its original purpose as a safety measure, AIS is increasingly being used in many other applications including fishing activities monitoring. AIS is an attractive data source due to its open protocol, high temporal resolution, and expanding regional regulations requiring more vessels to carry them. As an example that relates to our project both in application and geographical region, all fishing vessels of 15

meters or longer operating in the European Union are required to be fitted with an AIS device since May 2014 [28]; which applies to the majority of commercial fishing vessels in the region.

The open protocol of AIS signals means that not only the surrounding vessels, but anyone with an AIS receiver can access the information transmitted by vessels within range; including ground stations in coastal areas. The horizontal range of AIS signals is highly variable, and is influenced by the elevation of the receiver's antenna, which in effect determines the receiver's visible horizon (due to the Earth's curvature). Depending on the elevation and antenna height, ground stations in coastal areas are capable of receiving AIS signals usually only up to a range of about 40 NM (nautical miles) or 74 Kilometers [27]. In contrast, AIS signals travel much longer vertically and can reach the satellites in earth's orbit. Utilizing satellites to receive AIS signals is particularly useful in monitoring areas that are out of the range of coastal ground stations. The AIS data collected using ground stations is referred to as *coastal* or *terrestrial AIS*; and data collected using space-based receivers is called *Satellite AIS* or *S-AIS*. The AIS data used in this project is from terrestrial AIS, which has a more consistent and higher update rate than S-AIS.

2.2.2 Semantic modeling for fishing vessel trajectories

Trajectories

An object's raw location data captured by positioning devices, such as AIS, is a set of spatio-temporal points—i.e., timestamped location coordinates. The temporal sequence of the spatial points obtained by tracking an object can be referred to as a the object's *movement track* [70]. The movement track of an object can be captured throughout its existence, however, many applications are only interested in parts of that track. For example, in the case of tracking a ship's movements to monitor its fishing activities, the parts of the movement track that correspond to its mooring are of no significance. *Trajectories* are defined as parts of the movement track that are of interest in a particular application [70]. Two specific points on an object's movement track can identify the beginning and the end of a trajectory for that object [71]. Using the previous example, the beginning of a fishing vessel's trajectory can be considered when the ship leaves the port; and its subsequent arrival at the port can be the end.

Semantic enrichment

Trajectories are often complemented with some contextual information relating to the particular application. In our example, interpreting the trajectory of a vessel to infer its fishing activities requires some knowledge about the characteristics of fishing behaviour. *Semantic enrichment* is known as the process of supplementing trajectories with contextual information called *annotations* [57]. For example, segments of a fishing vessel's trajectory may be annotated with its activity: fishing, navigating, etc.

Depending on the application, segment annotations can be recorded manually (e.g. fisherman logging fishing activity), or captured automatically. The task of automatic annotation can be formulated as the automatic segmentation of the trajectory: breaking the trajectory into segments based on some homogeneity criteria. That is, to automatically determine segments of a trajectory that exhibit a consistent behaviour, and can be distinguished from other segments showing different behaviours. A more precise definition of such homogeneous segments, called *episodes*, is given as: "*a maximal continuous sub-sequence of a trajectory that adheres to certain criteria*", according to [52]. We are primarily interested in the automatic segmentation of vessel trajectories to determine fishing or non-fishing episodes.

Fishing episode identification

Trajectory segmentation is the process of identifying portions of a trajectory that are homogeneous by some measure. This subject is a broad area of study, most of which is not within scope of this thesis. Here we focus on techniques that apply to identifying fishing/non-fishing episodes of vessel trajectories.

Fishing activity identification techniques fall into the two following broad categories: (i) methods that are based on a set of predefined rules (using domain knowledge); and (ii) methods that do not use predefined criteria. The first category of methods consist mainly of methods that use speed-based rules to identify fishing activity [14, 33, 64, 25, 23, 53, 50]. Such methods are prevalent due to their simplicity, and effectiveness of vessel's speed as an indicator for fishing activity. In fact, most fishing vessels demonstrate a bi-modal speed distribution, where the two peaks correspond to fishing and cruising speeds [54, 8, 45]. The necessary domain knowledge for

speed-based methods is the operational speed range of the fishing gear; since it varies significantly among different types of gear (e.g. longliners, trawlers, etc.). Such domain knowledge is not always available, necessitating more sophisticated methods to detect fishing activity. The second category of methods is particularly valuable in the absence of domain knowledge. A number of such methods use some kind of similarity measure and various clustering techniques to group similar segments of a trajectory together [68, 78]. Others use techniques similar to temporal pattern recognition such as auto-regressive models [34], or Hidden Markov Models (HMM) [21].

Due the presence of detailed information about the types of fishing gear in our data, we used speed-based criteria to detect fishing episodes.

Semantic trajectory modeling

The concept of *trajectories* was previously presented in this section, followed by the introduction of *semantic enrichment* of the trajectories. Semantic enrichment was described as the process of supplementing the trajectories with contextual information; the result of which is called *semantic trajectories*. Several data models have been proposed to formalize the representation of semantic trajectories, each of which can be suitable for specific applications. For example, *stops* and *moves* model represents the trajectory as a series of stop and move episodes, where each episode can be annotated with semantic information [57]. This model can be used to represent tourist sightseeing routes; in which the tourists move from one attraction to another, stopping at each attraction for a period of time. A more general modelling approach, named CONSTANT (Conceptual Model of Semantic Trajectories), introduces the concept of *subtrajectories*, which can be generated based on the goals or the behaviour of the moving object, or the means of transportation [9]. More recently, the MASTER (Multiple Aspect Trajectories) model was proposed, which expands the previous works by not only providing a conceptual model, but also a logical schema in RDF (Resource Description Framework) [49]. As a result, the MASTER model accommodates for the enrichment of the trajectories with complex semantic objects, making it more flexible and expressive than the previous models.

In this study, the MASTER model will be used for the semantic modelling of the fishing vessel trajectories. The resulting set of trajectories constructed using the

MASTER models are referred to as *multiple aspect trajectories*. The MASTER model introduces the concept of *aspect*, which intends to distinguish between the semantic information that apply to the whole trajectory—called *long term* aspects—versus semantic information that apply to parts of the trajectories (or subtrajectories)—called *volatile* aspects [49]. This distinction is especially useful in our application for fishing vessel trajectories, as the semantic information in our case includes both kinds. For example, *long term aspects* would include the boat identification number (MMSI), fishing gear, etc.; which would not change during the course of the trajectory. On the other hand, *volatile aspects* would include information that can change during the trajectory and only apply to subtrajectories, such as speed and the activity of the vessel (fishing, navigating, etc.).

Chapter 3

From Raw Data to Spatio-Temporal Map of CPUE

Our goal in this chapter is to create a high resolution spatio-temporal map of CPUE, and then augment it with environmental data. The resulting spatio-temporal dataset will be used in the Chapter 4 for prediction analysis. The steps involved in the transformation and integration of the datasets, from their raw form to the format used for prediction analysis, represent a significant conceptual and practical part of this project.

To recapitulate, CPUE is an important index in fisheries science and management, which is used as a key indicator of sustainable harvesting. As noted in Section 2.1, creating a *spatio-tempral* CPUE map is not usually possible. That is because, in most cases the available data can only be used to calculate the CPUE over the whole study area, which results in a time-series dataset *without* a spatial dimension. The studies which we found that involve spatio-temporal CPUE, use very coarse temporal and spatial granularity. However, a rare combination of detailed and correlated data sources enabled us to create very high-resolution spatio-temporal CPUE maps, as will be explained in this chapter.

In Section 3.1, we describe the unique set of three data sources that motivated this project. Then, in Section 3.2, we explain the process of creating spatio-temporal maps of CPUE—assuming we are given a set of semantic trajectories that combine two of the data sources. The resulting CPUE maps are then augmented with environmental data. Finally, in Section 3.3, we backtrack to describe the construction of the semantic trajectories that we previously assumed were given, and were used to create the CPUE maps.

3.1 Data sources

The datasets used in this project, while all pertaining to the Northern Adriatic region, come from the following three different sources:

1. AIS data from fishing vessels
2. Daily landing reports (quantity and species of fish caught)
3. Environmental data

Each of the three data sources will be described in this section.

3.1.1 AIS data

The raw AIS dataset contains tracking data for the trawl fishing vessels operating in the North Adriatic sea. The dataset, which was provided by the Italian coast guard, covers the period of January 2015 to December 2016,[1] and contains over 62 million records of timestamped coordinates of the vessel locations. Each record consists of the boat identification number (MMSI), coordinates of the vessel location, and a timestamp. The AIS data is captured using land-based receivers, making it *terrestrial* AIS; which has very high temporal resolution with update rates up to every 2 seconds. This dataset is available as a result of the European Union regulation that requires all fishing vessels of 15 meters or longer to carry an AIS device since May 2014 [28]. The number of distinct vessels in the dataset are 70 for 2015 and 77 for 2016; all of which belong to the fishing fleet of Chioggia—one of the main fishing ports on the Italian coast of the North Adriatic sea. The vessels in the dataset are fitted with one of the four following trawl fishing gear: rapido (RAP), small bottom otter trawl (SOTB), large bottom otter trawl (LOTB), and midwater pair trawl (PTM). Each gear has a range of operational speed, which is presented in Table 3.3 and will later be used in the detection of vessel activities in section 3.3.

3.1.2 Landing data

The landing dataset refers to the daily landing reports—i.e., the quantity of catch in kilogram per species—for individual vessels, which was obtained from Chioggia's fish market. The dataset consists of records for daily landing of 104 fish species in 2015-2016. The dataset represents the landings from 17921 fishing trips carried out by 82 vessels during the two years. Each record in the dataset includes the boat identification number (MMSI), date of landing, the amount and the species of fish

[1] At the time of writing this thesis, new data for 2017 and 2018 has become available. This thesis is, however, limited to the 2015-2016 data.

caught. The boat identification number (MMSI) is used to associate the landing records with the AIS dataset as described in section 3.3. As a result, we can also determine the type of gear associated with each landing, which allows us to produce the breakdown of trip counts per each gear type, as shown in Table 3.1.

Exploratory analysis of the landing data reveals details such as the seasonality in the data and the share of most harvested species. Figure 3.1 shows the total monthly landings in 2015 and 2016, as well as the share of the five most harvested species. Comparison of the graphs for the two years indicate the presence of seasonality trends, such as lower catches in the winter and spring compared to the summer months. The month of August indicates zero catch for both years, reflecting the fishing ban that was in effect during that time. Figure 3.2 depicts the annual catch for the five most harvested species. It is also visible from both figures that 2015 landing amounts were higher than 2016.

Table 3.1. Number of trips per gear type (2015-2016)

Gear type	Number of fishing trips
Rapido (RAP)	8688
Large bottom trawler (LOTB)	3891
Mid-water pair trawl (PTM)	3872
Small bottom trawler (SOTB)	1179
Other	291
Total	17921

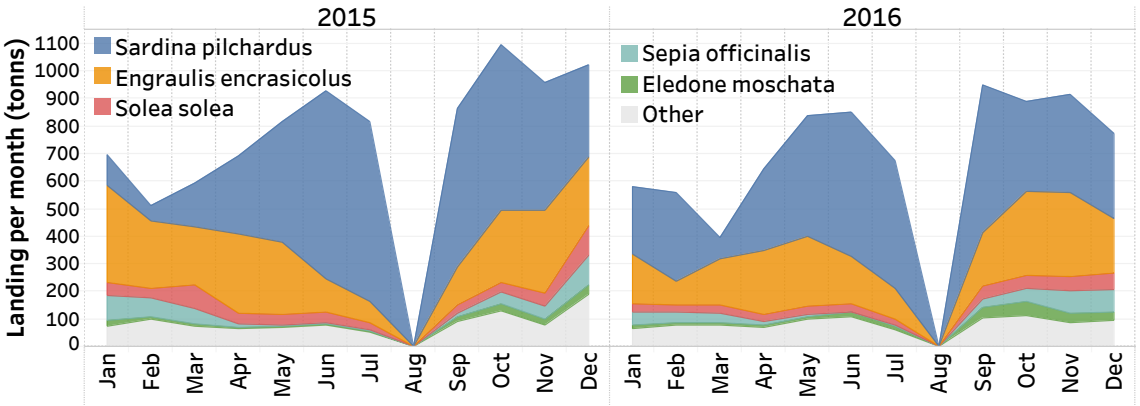

Figure 3.1. Chioggia's total monthly landing and species share in 2015-16

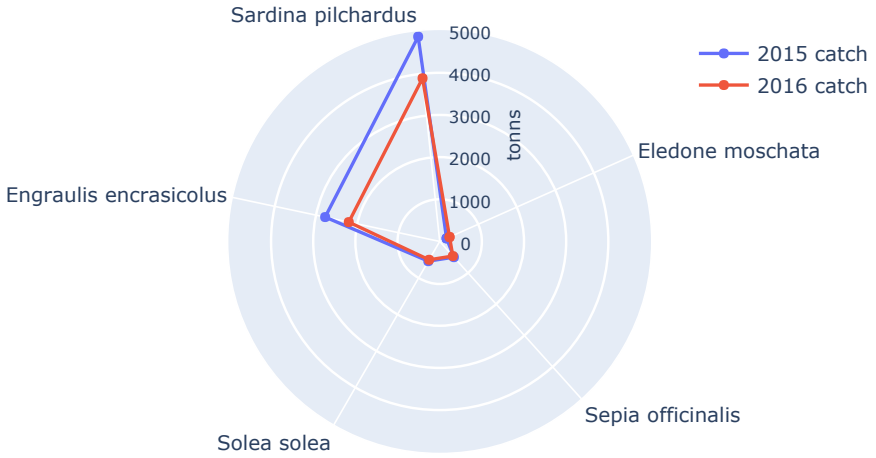

Figure 3.2. Annual landing per species in 2015-16

3.1.3 Environmental data

As noted in Section 2.1.2, environmental variables—such as sea surface temperature (sst), chlorophyll-a (chl-a) concentration, and wave height—can influence the CPUE. More specifically, sst and chl-a concentration can influence the species abundance and distribution, which in turn affects the catch rates. On the other hand, wave height is a measure of water turbulence which directly affects the fishing behaviour, and consequently catch rates. Therefore, considering such environmental factors could improve the accuracy of CPUE prediction. We utilize the three aforementioned environmental factors in our prediction analysis, which are obtained from satellite data [19].

3.2 Spatio-temporal dataset of CPUE and environmental factors

In this section we describe the steps involved in creating the spatio-temporal dataset of CPUE and environmental factors, which will later will be used in prediction modeling (Chapters 4 and 5). Subsection 3.2.1 describes the process of building the CPUE dataset using semantic trajectories, and the resulting dataset will be augmented with environmental factors in Subsection 3.2.2

As the name *Catch per Unit Effort* suggests, CPUE is calculated as the amount of '*catch*' divided by fishing '*effort*' ; where fishing effort quantifies the intensity of the fishing activity over a given *area* and *time period*. Therefore, to obtain a spatio-temporal map of CPUE, we first create separate maps of *catch* and *effort*.

As will be described in Section 3.2.1, obtaining a spatio-temporal map of effort requires knowing the *fishing* episodes of a vessel's trajectory; i.e., parts of the trajectory where the vessel was engaged in fishing activity. We would also need to know the amount of fish caught over each of those fishing episodes to create a spatio-temporal map of catches. The rare combination of the two data sources—detailed AIS dataset and the corresponding landing reports—enables us to infer the aforementioned pieces of required information. More specifically, the AIS data can be used to determine fishing episodes of vessel trajectories. Then, amount of catch can be associated with vessel trajectories by matching the MMSI and timestamps from the two data sources.

Such data can be represented by semantic trajectories; where the trajectories are enriched with semantic information such as vessel's *activity* (fishing or not fishing), and *amount of fish caught*. Moreover, a suitable semantic trajectory model would provide us with a means to unify spatial, temporal, and semantic features; to carry out complex queries; and to formulate the calculation of CPUE.

In Section 3.2.1, we formulate the derivation of spatio-temporal maps for *catch*, *effort*, and *CPUE* based on a given set of semantic trajectories. To do so, we assume that the trajectories are already constructed and given to us. However, the actual process of constructing the semantic trajectories is not trivial, and is later explained in detail in Section 3.3. There, we describe how the trajectories were built—using the AIS and landing reports datasets—by employing the MASTER [49] model.

3.2.1 Deriving spatio-temporal map of CPUE given the multiple aspect trajectories

Here we describe the process of generating spatio-temporal maps of *catch* and *effort* using a set of multiple aspect trajectories. From the two maps, we proceed to generate a high-resolution spatio-temporal map of CPUE, which is the objective of trajectory modeling and integration of the AIS and landing reports data sources.

In this section, we assume that a set of *multiple aspect trajectories* is previously constructed from the AIS and landing reports datasets. The given trajectories are constructed by following the MASTER model, in which they are enriched with *volatile* and *long-term* aspects. As noted in Section 2.2.2, *volatile* aspects are semantic information that can change over the duration of a trajectory, where as *long-term* aspects

do not. The constructed trajectories have the following specifications.

- each trajectory represents a unique fishing *trip* performed by a vessel; i.e., from when a vessel leaves the port for fishing, until the subsequent arrival at the port
- each trajectory is comprised of a sequence of smaller sections called *segments*
- volatile aspects pertain to each *segment* of a trajectory, and include *segment length*, *segment duration* ,*average speed on the segment*, *activity over segment (fishing, navigating, etc.)*, *amount of fish caught over segment (per species)*, etc.
- long-term aspects pertain to the whole trajectory, and include *MMSI*, *time of departure*, *time of arrival*, *type of fishing gear*, etc.

It is noteworthy that the above description specifies the information that the trajectories must contain for the purpose of creating the spatio-temporal CPUE dataset. However, as we will see in Section 3.3, the trajectories will contain additional semantic information that is necessary in their construction process, resulting in more complex trajectory objects.

Throughout this section, the set of all multiple aspect trajectories is denoted by T (set of all trips), and a single trajectory by t (a single trip).

Spatial grid

As mentioned previously, fishing effort, and thus CPUE, are defined over a spatial area. To calculate effort and CPUE with a high spatial granularity, we partition the study area into smaller square cells. The cells are created by imposing a spatial grid over the region of North Adriatic sea. Fishing effort and CPUE are then calculated over the individual cells. As described in later steps, utilizing the spatial grid enables us to take advantage of the detailed trajectories, and to produce a CPUE map with a high spatial resolution.

The size of the grid cells will determine the spatial granularity of the CPUE map. In theory, we can make the grid cells as small or as big as desired. However, care must be taken with the choice of cell size, as it profoundly affects the resulting CPUE dataset. For example, if the cell size is too small, then there is less chance that multiple vessel trajectories would fall in a particular cell. Therefore, the resulting dataset would consist of many cells with CPUE values obtained from a single, or very few trajectories. On the other hand, if the cell size is too large, it would not take

advantage of the availability of detailed vessel tracking data. We found that 5x5 *km* gird cells are suitable for our application. Other cell sizes such as .5x.5, 1x1, and 10x10 *km* were also considered. Figure 3.3 shows the 5x5 *km* grid imposed over the study area.

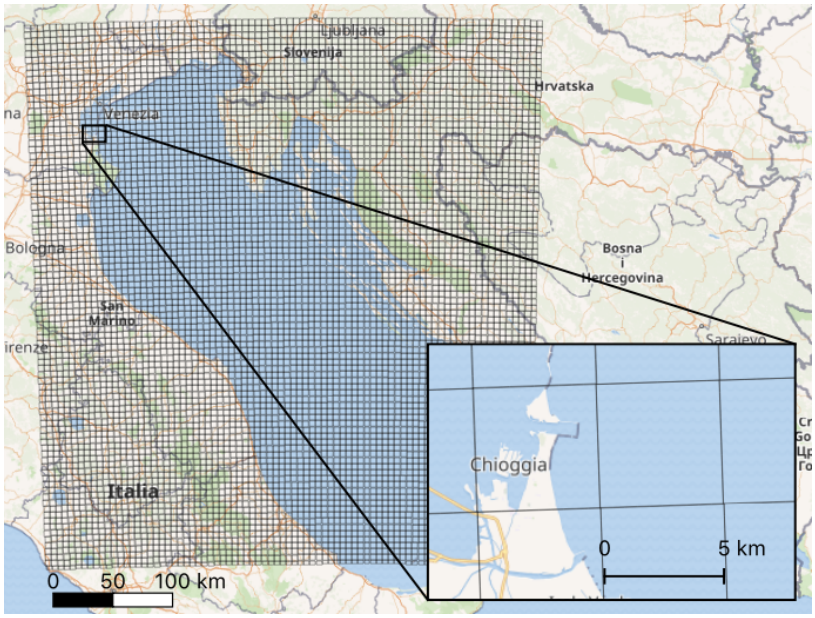

Figure 3.3. Spatial grid imposed on the North Adriatic region (5x5km).

Spatio-temporal map of catches

One of the two components needed in the calculation of CPUE is a spatio-temporal map of catches. That is, the amount of catch aggregated over the same spatial and temporal granularity chosen for CPUE—e.g., in a 24 hour period and over 5x5 *km* grid cells. The reason for the choice of cell size is explained in the previous section. Following a similar reasoning, the choice of temporal granularity would also affect the resulting dataset. If the time window is too short (e.g. hourly), then there is less chance that multiple vessels would be present in a particular cell within the same time window. Therefore, the resulting dataset would include many records that represent only a single, or very few vessels. On the other hand, if the time window is too long (e.g. weekly), then the effects of short-term environmental factors (e.g. weather) on catch rates would be lost.

Having a '*catch amount*' attribute attached to our trajectory segments, we calculate the aggregated catch by adding up the catch amounts of segments that fall into each grid cell, and have timestamps that are within the chosen time period. Maps of catches can be generated separately for each species, since segment's '*catch amount*' attributes are per species. Moreover, we use the trajectory's '*gear type*' attribute to consider the catches for each fishing gear separately. Grouping catches by gear type is done because different types of fishing gear differ in their efficiency for catching different species.

Based on the information above, we can compute the quantity of fish of a particular species s, caught in a cell c, during a time period p, by boats having a particular gear g.

Definition 1 Let c be a grid cell, p a time period, and g a gear, and s a species of fish; the *catch* for species s with respect to the gear g in cell c during the time period p is defined as follows:

$$catch(c, p, g, s) = \sum_{t \in T, gear(t)=g} quantity(t, c, p, s) \tag{3.1}$$

where

- T is the set of multiple aspect trajectories;
- $quantity(t, c, p, s)$ returns the sum of the catch quantities in kilograms for species s associated with the fishing segments of trajectory t that fall in cell c during period p.

As it will be described in detail in Section 3.3.5, there are two '*catch amount*' attributes attached to each trajectory segment: '*uniform catch*' and '*weighted catch*'. Because the *total* catch for each fishing trip is reported at the time of landing, it is required to distribute the catch amount over the fishing trajectory to obtain a '*catch amount*' attribute for each segment of the trajectory. The distribution of catch is performed in two ways, *uniform* and *weighted*, both of which will be discussed in Section 3.3.5. Consequently, the process described in this section results in the construction of two distinct maps of catch accordingly: *uniform* and *weighted*.

Spatio-temporal map of fishing effort

Fishing effort, or simply *effort*, is an important measure in fisheries science that indicates the intensity of fishing activities in the area of interest and over a certain

period of time. As the name suggests, fishing effort quantifies the "effort" vessels expend during their fishing activities.

Methods for quantifying effort vary greatly among fisheries due to the variety of techniques and gear that are employed [48]. For example, fishing effort for long-lines—which use hooks—and trawlers—which use nets—are not calculated the same way. Our dataset consists entirely of trawlers, which operate by dragging a trawl net when fishing. Therefore, our approach for calculating effort uses the length of the fishing segments from the vessel's (multiple aspect) trajectory.

We calculate the fishing effort over a time period and grid cell, similar to the construction of maps of catches described previously. In our approach, fishing effort is calculated as the area that is "*swept*" by the vessel inside a cell over a time period, divided by the area of the cell. The swept area is calculated as the total length of the fishing segments inside a cell over a time period, multiplied by the width of the trawl net. Therefore, this approach considers the specifications of the employed gear—in this case *gear width*—in the calculation of fishing effort. We used a constant gear width of $20m$ based on domain knowledge about the Chioggia's fishing fleet. Below is a more precise definition of the fishing effort for given grid cell, time period, and gear type.

Definition 2 Let c be a cell, p a time period, and g a gear. The *fishing effort* with respect to the gear g in the cell c during the time period p is defined as follows:

$$fe(c, p, g) = \frac{\left(\sum_{t \in T, gear(t)=g} len(t, c, p) \right) * gear\_width(g)}{area(c)} \tag{3.2}$$

where

- T is the set of multiple aspect trajectories;
- $len(t, c, p)$ returns the sum of the lengths of the fishing segments of trajectory t falling in cell c during time period p;
- $gear\_width(g)$ is the width of the net of gear g;
- $area(c)$ is the total area of the cell c.

Having the segments location and the '*segment length*' attribute attached to our trajectory segments, we can calculate the aggregated lengths of the fishing segments

that fall into each grid cell. As a result, a more realistic and accurate estimation of the swept area and fishing effort can be calculated for the choice of spatial granularity.

Spatio-temporal map of CPUE

Based on the maps of catch and effort created in the two previous steps, a definition of CPUE for a given grid cell, time period, gear type, and fish species is provided below.

Definition 3 Let c be a cell, p a time period, g a gear, and s a fish species; the *catch-per-unit-effort (CPUE)* for species s with respect to the gear g in cell c during the time period p is defined as follows:

$$cpue(c, p, g, s) = \frac{catch(c, p, g, s)}{fe(c, p, g)} \tag{3.3}$$

As noted previously, two distinct maps of catches are produced: one for *uniform* and one for *weighted* catch distribution. Accordingly, two distinct maps of CPUE are generated based on the two types of catch distributions. Figure 3.4 shows an example of the CPUE map for arbitrary p, g, and s.

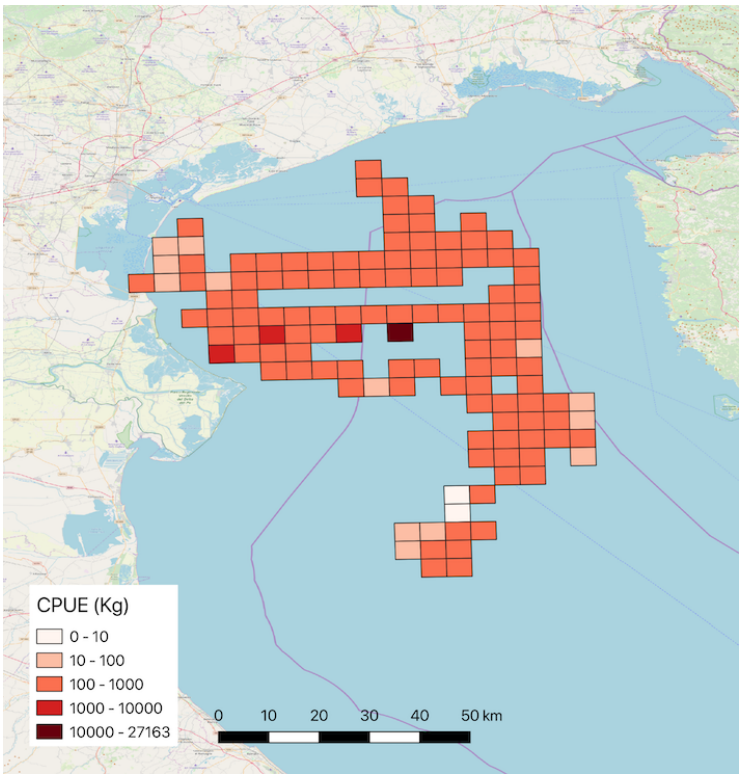

Figure 3.4. Map of CPUE (uniform distribution) for day 2015-02-02, gear *rapido*, and species *Sardina pilchardus*.

3.2.2 Augmenting the CPUE map with environmental data

As described in the previous section, the resulting CPUE maps are spatio-temporal datasets, were each record includes a spatial field (grid cell coordinates), a temporal field (time period), and the CPUE value for that grid cell and time period. Since the CPUE datasets will be used for prediction analysis, we augment them with environmental data that influence CPUE.

As noted in Section 3.1.3, sea surface temperature and chlorophyll-a concentrations are considered, since they affect species distribution. Wave height is also considered, as it influences fishing behaviour. All three datasets are resampled to match the temporal and spatial granularity of the CPUE maps. The resampled values are then concatenated to the CPUE datasets by matching the temporal and spatial fields of the records.

3.3 Construction of the multiple aspect trajectories

In this section we backtrack to outline the steps involved in transforming the raw AIS data into *multiple aspect semantic trajectories* following the MASTER [49] model (mentioned in Section 2.2.2), which were used to construct the spatio-temporal maps of catch, effort, and CPUE as described in Section 3.2.1. Since each step of the way describes the process of creating a piece that is used in the construction of semantic trajectories, the trajectory model as whole might get obscured by the detail. For that reason, we will provide a summary of the completed trajectory model at the end in Section 3.3.6.

3.3.1 Segment construction

The raw AIS data consists of a temporal sequence of spatial coordinates, where each point is labeled with a unique ship identification number (MMSI). A table containing the raw AIS data would have the following fields $\langle$*MMSI, timestamp, longitude, latitude*$\rangle$. The first step in our data transformation process is to convert the raw AIS data into a more convenient format for our analysis, namely *line segments* or simply *segments*. Segments are constructed by linear interpolation of pairs of temporally consecutive points (with identical MMSI) from the raw AIS table.

A benefit of working with line segments, instead of points, is that we can calculate and store semantic information related to each segment; such as segment length, duration, and average speed over that segment. The data fields of the segment table would include ⟨*MMSI, t0, t1, lon0, lat0, long1, lat1, segment_length, segment_duration, average_speed*⟩; where *t0*, *long0*, and *lat0* correspond to the timestamp, longitude, and latitude of the *start* point of a segment, and *t1*, *long1*, and *lat1* similarly correspond to the *end* point of the segment. Segment semantic attributes such as *segment_length*, *segment_duration*, and *average_speed* will be used in activity labeling and removing implausible trips.

Throughout our analysis, a *segment* represents the smallest unit of a vessel's movement. Semantic attributes that are not constant during the whole trajectory, but can change at the segments granularity level, are considered *volatile* aspects of the trajectory in MASTER model. Such attributes include *segment_length*, *segment_duration*, and *average_speed*. In contrast, *long-term* aspects are attributes that are constant during the whole trajectory. Such attributes include *MMSI* and the *fishing gear* of the vessel.

3.3.2 Activity labeling

Activity labeling of vessel movements is a cornerstone of our analysis, since calculating CPUE relies on the detection of vessel's fishing activity. Additionally, our trip identification algorithm is based on vessel activities as well. Vessel activity is attributed to each segment as an integer value between 0 and 4 (activity ID), denoting the activities listed in Table 3.2.

Table 3.2. Vessel activities

Vessel activity	Activity ID
in port	0
existing port	1
entering port	2
fishing	3
navigating	4

The '*in port*', '*exiting port*' and '*entering port*' activities are deduced from the position of the extremes of a segment with respect to Chioggia port's geographical

boundaries. Specifically, '*in port*' is identified as the situation when both *start* and *end* points of the segment are within the port's boundaries; '*exiting port*' is when the segment's *start* point falls inside and the *end* point falls outside the port boundaries; and '*entering port*' occurs when the reverse of the previous situation is true.

In the case that none of the previous situations occur, the '*fishing*' and '*navigating*' activities are identified by adopting a speed-based detection algorithm using the segment's average speed. This method employs domain knowledge about the fishing gear's operational speed ranges given in Table 3.3. More precisely, a segment is assumed to be a '*fishing*' episode when its average speed falls within the speed range of the corresponding gear; otherwise it is assumed to be a '*navigating*' episode.

Vessel activity is another *volatile* aspect of the trajectory in the MASTER model, since it is not constant during the whole trajectory. Figure 3.5 shows an example of a vessel trajectory with color coded segments based on its activity.

Table 3.3. Fishing gear speed ranges

Gear name	Gear ID	Min speed (km/h)	Max speed (km/h)
Rapido	RAP	7.408	12.964
Small bottom otter trawl	SOTB	3.704	8.334
Large bottom otter trawl	LOTB	3.704	8.334
Midwater pair trawl	PTM	3.704	10.186

3.3.3 Trip identification

Up to this point, we have constructed the segments, and supplemented them with an activity attribute. For the purpose of calculating CPUE, we would also need to attach the *catch amount* to the *fishing* segments. Therefore, we need to associate the segments—constructed using the AIS data—with the landing reports dataset.

Each record in the landing reports dataset corresponds to the amount of catch reported after a vessel completes a fishing *trip*. A fishing *trip* is considered to be the trajectory of a vessel between the time it leaves the port, and its subsequent arrival at the port. However, we have an unbroken sequence of segments without any indication of the individual trips. Therefore, to associate the AIS and landing reports datasets, we would need to group the segments into distinct fishing *trips*. All segments in a

group can then be labeled with a trip identification number. It naturally follows that an individual fishing *trip* can be considered a unique *trajectory* in our application.

We characterize a trip as a sequence of consecutive segments, where the activity label of the first segment is '*exiting port*' (activity ID 1), and the last segment's activity label is '*entering port*' (activity ID 2). Then, each trip can be enriched with *long-term* aspects such as total duration of the trip (*trip_duration*), total length of the trip (*trip_length*), and total fishing length of the trip (*trip_fishing_length*). We use these attributes to enforce the following constraints on the trips: it has to last at least 1 hour, and have a minimum length of 2 *km*, from which at least 100 *m* have to be classified as fishing.

The time of exiting port is then added as an attribute to all segments in the corresponding trip, and referred to as *departure_time*. As a result, the combination of *MMSI* and and *departure_time* identifies all segments belonging to a unique trip. Similar to *MMSI*, *departure_time* is also a *long-term* aspect of the trajectory.

Figure 3.5 shows an example of a uniquely identified fishing trip. Part of the trajectory is magnified to show the first and last segments of the trip, when the vessel is '*exiting port*' (activity ID 1), and '*entering port*' (activity ID 2).

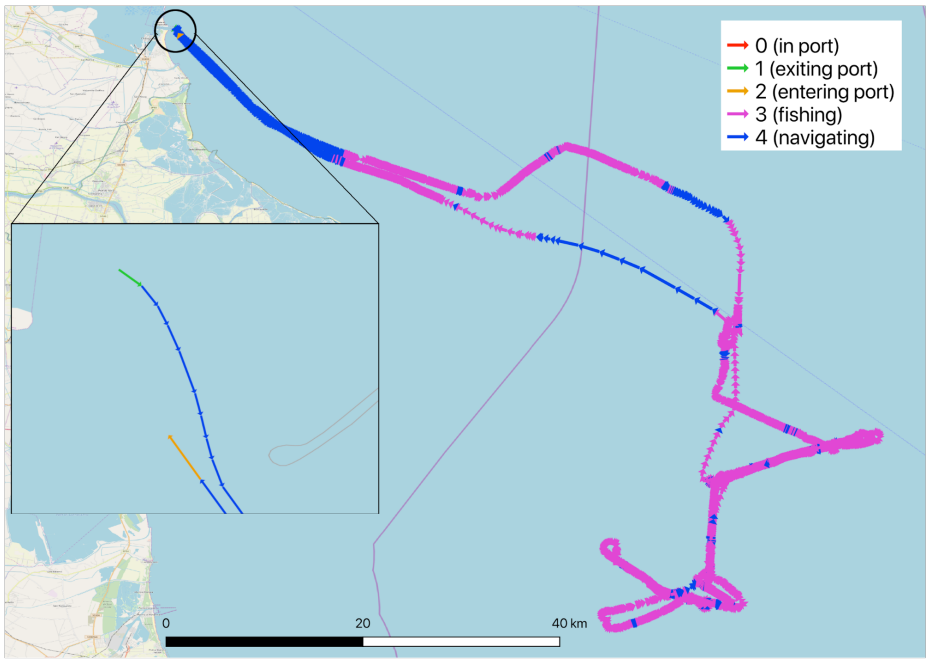

Figure 3.5. Example of a fishing trip trajectory with activity IDs. Trip information — date: 2014-04-15; gear: RAP; total catch: 399kg.

3.3.4 Assigning landing reports to trips

The daily landing reports dataset obtained from the Chioggia's fish market consists of detailed information about each trading transaction; including the landing date, MMSI of the fishing vessel, the species caught, and the quantity of catch per species. The goal here is to associate each landing report with a fishing trip obtained from the previous step.

To accomplish this task, for each landing report, we attempt to find the vessel's trip with the arrival time at port that is consistent with the recorded landing date. In this step, it is important to take into account the operating hours of the fishing market at Chioggia port. If a vessel's arrival time at port is too late in the day, the trading transaction would be done in the following day. Therefore, we associate a landing report with the vessel's trip that has the most recent arrival in the port before 4 PM of the landing date. Arrivals after 4 PM are associated with transactions occurring the next day. The amount of catch reported in the caption of Figure 3.5 was obtained by associating the fishing trip with the corresponding record in the landing reports dataset.

3.3.5 Distribution of catches over trips

Here we describe the two approaches we used to distribute the catch amounts over the assigned fishing trips. This step is essential in creating a spatio-temporal map of catch amounts, which is necessary to create the high resolution spatio-temporal CPUE dataset. The result of this step is used in Formula 1 to construct the spatio-temporal map of catches as described in Section 3.2.1.

We employ the two following techniques for distributing catch amounts over trips

- *uniform* distribution, or
- *weighted* distribution.

In the first approach, for each landing report, the amount of fish is *uniformly* distributed along the *fishing* segments of the corresponding trip. More precisely, each *fishing* segment of the trip is attributed with a portion of the total amount of catch proportional to the segment's length. This method is an effective but basic

approach; and we acknowledge that the assumption of uniform catch distribution is an oversimplification of reality.

In the second approach, we attempt to improve the first method by performing a *weighted* distribution of catches. The idea behind this approach is that the areas where more vessels are fishing during a given time period, are more likely to have higher catch rates. Therefore, the segments of a vessel's trip that fall in areas with more fishing activities would have higher weights in the catch distribution.

To quantify the fishing activities, we created a heat-map of the count of the vessels engaged in fishing activities on the spatial grid. The grid is the same one that is used for calculating CPUE, and is described in Section 3.2.1. The heat-map is created by counting the number of unique vessels that were engaged in fishing activities inside each grid cell over a 24 hour period. Each fishing segment is then attributed with a weight, which is equal to the vessel count in the heat-map cell containing that segment. Finally, the weighted distribution is performed by attributing each fishing segment with the portion of the total amount of catch that is proportional to its length multiplied by the segment's weight.

The uniform and weighted catch are added to each segment as attributes, and are considered *volatile* aspects of the trajectory. Moreover, both methods are performed for all the species in the landing reports separately, and result in distinct attributes for each species. This level of detail gives us the choice to consider the species separately. Of course, we have the option to aggregate them, if the species information is not relevant to the study.

3.3.6 Summary of the trajectory model

The preceding steps explained the piece by piece construction of semantic trajectories following the MASTER model [49]. Here we describe resulting trajectory model as a whole picture.

We defined a *trajectory* as a sequence of *segments* that represent a distinct fishing *trip*. We chose line *segments*, instead of spatio-temporal points, as the minimum granularity to attach semantic information; due to the convenience they provide in detecting homogeneous trajectory portions for activity labeling. The trajectories

were enriched with semantic information, which are categorized as *volatile* or *long-term aspects* in the *multiple aspect trajectory* (MASTER) model [49]. *Volatile* aspects are attached to *segments*, and can change during the *trajectory* (trip); whereas *long-term* aspects are constant during the whole trajectory [49]. Both types of semantic attributes used in our trajectory model are listed below.

- *long-term aspects*: MMSI, departure time of the trip, departure port, type of fishing gear, duration of the trip, total length of the trip, fishing length of the trip, total amount of fish caught in the trip
- *volatile aspects*: segment length, segment duration, segment average speed, activity of the boat, amount of fish caught on the segment (separate attributes for all species, each with both values for uniform or weighted distribution methods)

By using the MASTER model we are able to represent different aspects of our trajectories in a uniform and simple way. Moreover, this representation allows us to perform complex queries merging together spatial, temporal and semantic features.

Chapter 4

Prediction Modelling: Problem and Methods

As mentioned previously, the final goal of this study is to perform a spatio-temporal forecast (prediction) of CPUE using the spatio-temporal dataset created in Section 3.2—i.e. the spatio-temporal map of CPUE augmented with environmental variables. In this chapter we state the specifics of the prediction problem, describe our approach to perform the prediction task using regression modeling, and finally specify our model evaluation methods.

As pointed out in Section 3.1.2, the landing dataset contains catch amounts for various species caught using various types of gear. However, we have limited the prediction modeling to one species and one gear type. More specifically, *Sardina pilchardus* is chosen as the species for prediction modelling, since it constitutes the largest share of annual catches (see Figure 3.2). Similarly, the fishing gear *rapido* (RAP) is chosen, since it makes up almost half of the fishing trips (see Table 3.1).

4.1 Prediction task

Employing the spatial grid that was used to create the CPUE map in Section 3.2.1, the task of *spatio-temporal* prediction of CPUE can be described as predicting CPUE values for individual grid *cells* and over a specific time *period*. As mentioned in Section 2.1.1, in fisheries science and management, CPUE is often obtained for relatively long time periods, such as annual, seasonal, or monthly. Since our dataset only spans over two years, one of which is going to be used to train the prediction model, we decided to perform prediction for the shortest conventional time period—i.e. *monthly*. Given this information, the problem can be stated as follows.

Problem statement Given a two-year spatio-temporal dataset consisting of *daily* CPUE values for individual spatial cells augmented with *daily* environmental factors, build and evaluate a model to predict *monthly* CPUE values for each spatial cell.[1]

[1]Spatial cells are from the spatial grid described in Section 3.2.1.

Prediction for shorter time periods (e.g. daily or shorter) is not suitable for fisheries management. Besides, the extreme variance in CPUE values over a short time period and a small area (e.g. a grid cell) makes prediction impractical. Prediction for longer time periods (e.g. seasonal or annual) would be desirable, but not feasible with our available dataset which only spans over two years. A dataset with longer time span would be required to capture seasonal and annual trends in CPUE.

4.1.1 Regression modelling

Since CPUE is real-valued and continuous, the prediction task can be formulated as a regression problem; where the output of the regression model is the CPUE value for time period (i.e., month) p and grid cell c. The model is trained on data from 2015, and evaluated on 2016. This method for splitting the data is chosen due to the sequential nature of the data, for which random sampling is not suitable. Furthermore, to evaluate our monthly prediction results on all months of the year, the data for the whole year of 2016 is set aside for evaluation. The inputs to the model (model features or attributes) are environmental, spatial, and temporal attributes as presented in Table 4.1.

Table 4.1. Model attributes

	Attribute description	Symbol	Type	Unit
Environmental	daily chlorophyll-a concentration	chl	float	mg/mˆ3
	daily sea surface temperature	sst	float	kelvin
	daily spectral significant wave height	vhm0	float	meter
Spatial	latitude of grid cell centre	lat	float	degree
	longitude of grid cell centre	lon	float	degree
Temporal	day of year (1-365)	doy	int	
	month of year (1-12)	moy	int	
	week of year (1-53)	woy	int	
	season (1-4)	season	int	

The machine learning algorithms used for regression modelling will be discussed in Chapter 5.

4.1.2 Adjusting temporal granularity

In contrast with the goal of our forecast modelling, which is the prediction of *monthly* CPUE, the environmental factors (e.g., wave height) can affect fishing activities on a daily basis. Monthly aggregation of the environmental attributes prior to modelling could lead to loss of the information regarding their daily effects on the fishing activities. We attempt to preserve that information by deferring the aggregation to after the regression modelling, which is explained as follows. First, *daily* training data from 2015 is used to train a regression model that subsequently is used to produce *daily* predictions for individual grid cells for the year 2016. Then, the *monthly* mean (average) of *daily* CPUE predictions for each cell is calculated to obtain the *monthly* CPUE forecast for the respective cell. Averaging is used to aggregate the predicted CPUE values, as opposed to summation, because CPUE values are ratios ($CPUE = catch/effort$) and do not produce a meaningful sum (as opposed to catch amounts, for example).

The model output is considered to be the average monthly predictions obtained from the method that was described above. The model evaluation is performed against average *monthly* CPUE values that were calculated in the same fashion, but using the actual 2016 data. Equation (4.1) shows the calculation of actual and predicted average CPUE for a given cell c over a given period p (i.e. month),[2] respectively denoted by $y_{c,p}$ and $\hat{y}_{c,p}$;

$$y_{c,p} = \frac{1}{|D_{c,p}|} \sum_{d \in D_{c,p}} cpue(c,d) \quad \text{and} \quad \hat{y}_{c,p} = \frac{1}{|D_{c,p}|} \sum_{d \in D_{c,p}} \widehat{cpue}(c,d) \tag{4.1}$$

where $D_{c,p}$ indicates the set of all days d in period p for which cell c has a CPUE value. $cpue(c,d)$ and $\widehat{cpue}(c,d)$ are respectively the actual and predicted daily CPUE values for cell c on day d.

[2] As pointed out in the beginning of Chapter 4, we consider CPUE only for the gear *rapido* and species *Sardina pilchardus*. Hence, instead of writing $cpue(c, p, rapido, Sardina\ pilchardus)$, we simply use $cpue(c,p)$, omitting the gear and species names.

4.2 Model evaluation

Baseline model

The baseline, which is used as the benchmark to compare with our prediction models, is to simply use the last observed value as the forecast. Using this baseline is standard practice in forecast modeling, which is referred to as *naïve forecast* [40]. Comparing the forecast models against this baseline—rather than comparing to each other—has the added advantage of providing a consistent benchmark that is independent of the choice of the regression model. In particular, for this experiment, the baseline average monthly prediction of CPUE for a given cell and month in 2016 is the respective average monthly CPUE for that cell and month from 2015. The baseline prediction is shown in Equation (4.2).

$$\hat{y}^{*}_{c,p} = y_{c,p\downarrow 1} \tag{4.2}$$

where $y_{c,p\downarrow 1}$ is the actual value for cell c at the same period moved one year backward. For instance, if $p = \textit{June2016}$ then $p \downarrow 1 = \textit{June2015}$. In the scenario that a particular cell is present (has some fishing activity) for a given period in 2016, but is absent (has no fishing activity) for the same period in 2015, the baseline prediction is considered to be zero.

Evaluation metrics

The metrics used for model evaluation are as follows.

- Mean Absolute Error (MAE) is calculated for each period p (i.e., month) as the mean of absolute errors of the predicted average CPUE for all cells in that period. MAE for period p is shown in Equation (4.3);

$$MAE_p = \frac{1}{|C_p|} \sum_{c \in C_p} |\hat{y}_{c,p} - y_{c,p}| \tag{4.3}$$

  where C_p denotes the set of all cells with a CPUE value in period p; and $\hat{y}_{c,p}$ and $y_{c,p}$ are respectively predicted and actual CPUE values for cell c and period p. MAE for the baseline prediction is calculated similarly and it is denoted by MAE^{*}.

- Normalized Mean Absolute Error (nMAE) is calculated for each period as the MAE for that period divided by the mean of the actual CPUE for that period. nMAE for period p is shown in Equation (4.4).

$$nMAE_p = \frac{MAE_p}{\mu_p} \quad ; \quad \mu_p = \frac{1}{|C_p|} \sum_{c \in C_p} y_{c,p} \tag{4.4}$$

  The advantage of this metric over MAE is that it adjusts for the magnitude of the mean of CPUE in a time period, providing a metric that is comparable among different periods. Besides, it indicates the size of MAE relative to the mean.

- Relative Absolute Error (RAE) is a measure of model performance relative to the baseline model; it is calculated as the ratio of model MAE to the baseline MAE* for a given period [2], as shown in Equation (4.5).

$$RAE_p = \frac{MAE_p}{MAE_p^*} \tag{4.5}$$

  RAE provides a quick way to determine how a model performs compared to the baseline model. RAE values of less than 1 indicate that the model is performing better than the baseline, and values greater than 1 indicate that the model is performing worse than the baseline.

Chapter 5

Prediction Modelling: Experiments and Results

As noted in Section 4.1.1, our approach to predict monthly CPUE involves training a regression model to predict daily CPUE. [1] In this chapter we discuss the machine learning methods that are used for regression modeling, and then report the results and performance metrics for each method.

5.1 Experiments

Regression models are built using the spatio-temporal dataset consisting of the daily CPUE and environmental variables pertaining to individual grid cells. The response (or target) variable of regression is the CPUE value, and the input features to the models are environmental, spatial, and temporal variables as described in Section 4.1.1. The only categorical feature, i.e. season with four possible values, was converted to four binary features using one-hot encoding.

Models are trained on data from 2015, and evaluated on 2016. This method of partitioning the data is chosen over other sampling methods (e.g. cross-validation) because of the sequential nature of the dataset. More specifically, this method ensures that the training data contains records from all time periods in a year (e.g. seasons, months, etc.). Furthermore, this partitioning strategy simulates the way that data would be available for forecast modeling in the industry. That is, predictive models would be trained using historical data as it becomes available.

As noted in Section 3.2.1, two distinct CPUE datasets were created based on the two catch distribution methods: *uniform* and *weighted* catch distributions. Separate prediction models are built for the two CPUE datasets, and results are reported for both.

[1] As described in Section 4.1.2 monthly CPUE for each cell is calculated by taking the average of daily predictions for that cell over the month.

5.1.1 Machine learning methods

In this section, we discuss the machine learning methods employed for regression modeling, and reference the particular software implementations that were utilized. Four machine learning methods were used, each of which was part of either of two software packages: H2O [37] or Scikit-learn [61]. H2O is an open-source distributed machine learning platform created by H2O.ai; and Scikit-learn is an open-source machine learning library for Python.

Model hyperparameters for each method were chosen by performing grid searches on the hyperparameter space and taking the best scoring settings. Specifically, a Cartesian grid search was performed on the combination of different values for hyperparameters, and the values that resulted in the lowest mean absolute error on the validation dataset were selected. Choice of hyperparameters for each model is reported in Appendix A. The four machine learning methods are described below.

Generalized Linear Model (GLM)

Regularized linear regression was used as one of the regression methods. In particular, we made use of the Generalized Linear Model (GLM) module from H2O [37]. H2O GLM supports elastic net regularization, which is a combination of L1 (LASSO regression) and L2 (Ridge regression) regularization methods. It also supports the Tweedie family of distributions, which include normal, Poison, gamma, and their combinations.

The Tweedie model with elastic net regularization was chosen for regression modeling because it is flexible with respect to the data distribution and resists over-fitting by penalizing model complexity. The hyperparameteres that parameterize Tweedie distribution and elastic net regularization are chosen through grid search, and reported in Appendix A.

SVM

Similar to Support Vector Machine (SVM) for classification, SVM for regression uses the kernel trick to transform the data into a high dimensional space [75]. SVM can

model nonlinearities by using a nonlinear kernel, such as polynomial or radial basis function (RBF) kernels. At the time of writing this document, H2O's [37] implementation of SVM is only for binary classification problems, and not regression. Therefore, the SVR module from Scikit-learn [61] was used for regression modeling, which is an SVM implementation for regression based on LIBSVM [13]. Model hyperparameters, including the type of kernel, were chosen via grid search and are reported in Appendix A.

XGBoost

Extreme Gradient Boosting (XGBoost) is an ensemble tree-based machine learning algorithm which relies on tree boosting [15]. Ensemble learning methods can improve accuracy, and reduce bias and variance by combining outputs of many base learners [24]. Tree boosting algorithms, and XGBoost in particular, have recently been popular due to their success in a number of machine learning competitions [15]. XGBoost applies the boosting technique described in [31] to decision trees as its base learners. More specifically, it sequentially builds decision tree models by training on versions of the data that are re-weighted based on results from the previous models such that the new model performance is improved. Then an aggregation of outputs of all the models is considered to be the final output.

XGboost was chosen as one of the regression methods because of its desirable properties, such as no assumption on data distribution, being able to model nonlinearities, and resistance to over-fitting. We used the H2O [37] XGBoost module. Hyperparameters were chosen via grid search and are reported in Appendix A.

Random Forests

Random Forests (RF), similar to XGBoost, is an ensemble tree-based machine learning algorithm [11]. When used for regression, the RF model output is calculated as the average of outputs of many individual regression trees. The regression trees are trained independent of each other using an ensemble learning method called bootstrap aggregating (Bagging). In Bagging, the base learners are trained on subsets of the dataset that are chosen using random sampling with replacement—i.e., bootstrap samples [10]. At each node of the individual trees, one attribute is chosen

from a randomly selected subset of all attributes. Bagging and the randomization of the attribute selection process are effective generalization methods that result in the robustness of RF against overfitting.

RF has similar desirable properties to other ensemble tree-based methods (e.g. XGBoost), such as no assumption on data distribution, being able to model nonlinearities, and resistance to over-fitting. Therefore, RF is chosen as one of the regression methods in this study. The H2O [37] DRF (Distributed Random Forests) module was employed, and model hyperparameters are chosen via grid search and reported in Appendix A.

5.2 Results and Discussion

In Section 5.2.1 we present the model performance metrics and a summary of the results for the four aforementioned machine learning methods. Then we proceed to provide interpretation and discussion of the results in Section 5.2.2.

5.2.1 Results

Tables 5.1, 5.2, 5.3, and 5.4 present evaluation metrics for monthly CPUE prediction models built using RF, XGBoost, SVM, and GLM respectively. Each table reports the three evaluation metrics that were described in Section 4.2 (Mean Absolute Error (MAE), Normalized Mean Absolute Error (nMAE), and Relative Absolute Error (RAE)), as well as the mean of CPUE to provide context for the magnitude of MAE. Defined by Formula 4.4, nMAE allows for a meaningful comparison of model performance on different time periods (e.g. months) by normalizing MAE with the mean CPUE of that period. It also provides a measure of the magnitude of MAE relative to the mean. On the other hand, RAE—defined by Formula 4.5—provides a measure of the magnitude of MAE relative to the baseline MAE; where RAE<1 indicates that the model MAE is less than the baseline MAE, and vice versa.

Each table is divided into two horizontal sections indicated by *uniform* and *weighted* catch distributions. The two sections respectively contain the results pertaining to the model built using the CPUE dataset obtained from *uniform* or *weighted* catch distributions as described in Section 3.2.1. In each table, the first 12 rows for each type of distribution (*uniform* or *weighted*) pertain to the 12 months of 2016; and

the 13^{th} row (labeled 'All') shows the annually averaged metrics. More specifically, p (time period)—in the evaluation metrics formulas in Section 4.2—is set to the particular month in 2016 for the monthly metrics; while it is set to the entire year for the annually averaged metrics.

Table 5.1. Random Forests evaluation metrics for monthly CPUE

	Month (2016)	MAE RF	MAE baseline	CPUE mean (actual)	nMAE RF	nMAE baseline	RAE
Uniform catch distribution	1	2065.28	2473.68	2423.60	0.85	1.02	0.83
	2	1758.83	2473.26	1728.77	1.02	1.43	0.71
	3	812.98	881.96	983.87	0.83	0.90	0.92
	4	622.05	663.76	732.72	0.85	0.91	0.94
	5	862.11	948.41	936.90	0.92	1.01	0.91
	6	675.91	886.13	815.72	0.83	1.09	0.76
	7	2333.37	2377.42	2419.54	0.96	0.98	0.98
	8†	na	na	na	na	na	na
	9	3078.92	3168.13	3379.25	0.91	0.94	0.97
	10	1430.51	1705.33	1733.98	0.82	0.98	0.84
	11	2101.59	2295.61	2372.60	0.89	0.97	0.92
	12	1113.45	1213.80	1237.16	0.90	0.98	0.92
	All	1490.18	1707.45	1658.45	0.90	1.03	0.87
Weighted catch distribution	1	1947.78	2188.87	2193.30	0.89	1.00	0.89
	2	1474.28	1963.45	1593.04	0.93	1.23	0.75
	3	658.65	740.02	782.88	0.84	0.95	0.89
	4	441.24	486.75	529.31	0.83	0.92	0.91
	5	580.32	643.32	641.55	0.90	1.00	0.90
	6	436.84	637.21	542.09	0.81	1.18	0.69
	7	1474.76	1491.21	1523.94	0.97	0.98	0.99
	8†	na	na	na	na	na	na
	9	2239.63	2336.55	2461.78	0.91	0.95	0.96
	10	1124.27	1242.47	1312.44	0.86	0.95	0.90
	11	1297.63	1660.43	1526.47	0.85	1.09	0.78
	12	778.46	876.93	868.05	0.90	1.01	0.89
	All	1116.69	1290.72	1254.27	0.89	1.03	0.87

† No data available due to the fishing ban.

Table 5.2. XGBoost evaluation metrics for monthly CPUE

	Month (2016)	MAE XGBoost	MAE baseline	CPUE mean (actual)	nMAE XGBoost	nMAE baseline	RAE
Uniform catch distribution	1	2249.65	2473.68	2423.60	0.93	1.02	0.91
	2	1522.91	2473.26	1728.77	0.88	1.43	0.62
	3	863.99	881.96	983.87	0.88	0.90	0.98
	4	635.94	663.76	732.72	0.87	0.91	0.96
	5	842.25	948.41	936.90	0.90	1.01	0.89
	6	708.75	886.13	815.72	0.87	1.09	0.80
	7	2337.65	2377.42	2419.54	0.97	0.98	0.98
	8†	na	na	na	na	na	na
	9	3269.45	3168.13	3379.25	0.97	0.94	1.03
	10	1590.81	1705.33	1733.98	0.92	0.98	0.93
	11	2191.33	2295.61	2372.60	0.92	0.97	0.95
	12	1072.81	1213.80	1237.16	0.87	0.98	0.88
	All	1521.52	1707.45	1658.45	0.92	1.03	0.89
Weighted catch distribution	1	2065.89	2188.87	2193.30	0.94	1.00	0.94
	2	1445.94	1963.45	1593.04	0.91	1.23	0.74
	3	685.52	740.02	782.88	0.88	0.95	0.93
	4	447.53	486.75	529.31	0.85	0.92	0.92
	5	568.17	643.32	641.55	0.89	1.00	0.88
	6	456.02	637.21	542.09	0.84	1.18	0.72
	7	1466.03	1491.21	1523.94	0.96	0.98	0.98
	8†	na	na	na	na	na	na
	9	2363.72	2336.55	2461.78	0.96	0.95	1.01
	10	1203.82	1242.47	1312.44	0.92	0.95	0.97
	11	1380.24	1660.43	1526.47	0.90	1.09	0.83
	12	741.98	876.93	868.05	0.85	1.01	0.85
	All	1148.23	1290.72	1254.27	0.92	1.03	0.89

† No data available due to the fishing ban.

Table 5.3. SVM evaluation metrics for monthly CPUE

	Month (2016)	MAE SVM	MAE baseline	CPUE mean (actual)	nMAE SVM	nMAE baseline	RAE
Uniform catch distribution	1	2233.32	2473.68	2423.60	0.92	1.02	0.90
	2	1554.33	2473.26	1728.77	0.90	1.43	0.63
	3	846.10	881.96	983.87	0.86	0.90	0.96
	4	625.52	663.76	732.72	0.85	0.91	0.94
	5	835.09	948.41	936.90	0.89	1.01	0.88
	6	723.93	886.13	815.72	0.89	1.09	0.82
	7	2335.79	2377.42	2419.54	0.97	0.98	0.98
	8†	na	na	na	na	na	na
	9	3232.64	3168.13	3379.25	0.96	0.94	1.02
	10	1588.87	1705.33	1733.98	0.92	0.98	0.93
	11	2199.28	2295.61	2372.60	0.93	0.97	0.96
	12	1064.26	1213.80	1237.16	0.86	0.98	0.88
	All	1518.04	1707.45	1658.45	0.92	1.03	0.89
Weighted catch distribution	1	2068.22	2188.87	2193.30	0.94	1.00	0.94
	2	1477.00	1963.45	1593.04	0.93	1.23	0.75
	3	691.87	740.02	782.88	0.88	0.95	0.93
	4	455.27	486.75	529.31	0.86	0.92	0.94
	5	572.28	643.32	641.55	0.89	1.00	0.89
	6	475.17	637.21	542.09	0.88	1.18	0.75
	7	1469.71	1491.21	1523.94	0.96	0.98	0.99
	8†	na	na	na	na	na	na
	9	2361.24	2336.55	2461.78	0.96	0.95	1.01
	10	1217.63	1242.47	1312.44	0.93	0.95	0.98
	11	1403.84	1660.43	1526.47	0.92	1.09	0.85
	12	749.66	876.93	868.05	0.86	1.01	0.85
	All	1159.26	1290.72	1254.27	0.92	1.03	0.90

† No data available due to the fishing ban.

Table 5.4. GLM evaluation metrics for monthly CPUE

	Month (2016)	MAE GLM	MAE baseline	CPUE mean (actual)	nMAE GLM	nMAE baseline	RAE
Uniform catch distribution	1	2228.55	2473.68	2423.60	0.92	1.02	0.90
	2	1584.97	2473.26	1728.77	0.92	1.43	0.64
	3	940.59	881.96	983.87	0.96	0.90	1.07
	4	719.38	663.76	732.72	0.98	0.91	1.08
	5	906.55	948.41	936.90	0.97	1.01	0.96
	6	723.42	886.13	815.72	0.89	1.09	0.82
	7	2357.29	2377.42	2419.54	0.97	0.98	0.99
	8†	na	na	na	na	na	na
	9	3143.28	3168.13	3379.25	0.93	0.94	0.99
	10	1531.83	1705.33	1733.98	0.88	0.98	0.90
	11	2133.73	2295.61	2372.60	0.90	0.97	0.93
	12	1080.01	1213.80	1237.16	0.87	0.98	0.89
	All	1533.19	1707.45	1658.45	0.92	1.03	0.90
Weighted catch distribution	1	2043.92	2188.87	2193.30	0.93	1.00	0.93
	2	1483.81	1963.45	1593.04	0.93	1.23	0.76
	3	761.41	740.02	782.88	0.97	0.95	1.03
	4	517.49	486.75	529.31	0.98	0.92	1.06
	5	622.20	643.32	641.55	0.97	1.00	0.97
	6	479.77	637.21	542.09	0.89	1.18	0.75
	7	1499.37	1491.21	1523.94	0.98	0.98	1.01
	8†	na	na	na	na	na	na
	9	2296.93	2336.55	2461.78	0.93	0.95	0.98
	10	1198.79	1242.47	1312.44	0.91	0.95	0.96
	11	1367.93	1660.43	1526.47	0.90	1.09	0.82
	12	765.68	876.93	868.05	0.88	1.01	0.87
	All	1170.29	1290.72	1254.27	0.93	1.03	0.91

† No data available due to the fishing ban.

5.2.2 Discussion

Magnitude of MAE

The first noticeable trait in the results, evident from tables 5.1 to 5.4, is that MAE (Mean Absolute Error) for all the models and the CPUE mean are fairly close in magnitude. This trait can also be observed from the nMAE (normalized MAE) values, which are consistently close to 1 for all models (between .81 and 1.02). The relatively large MAE is an indicative of the limitations of the models, which will be discussed in Chapter 6. However, regardless of the magnitude of MAE, the models that outperform the baseline prediction are still of value.

Model performance vs. baseline

To see whether the models provide an improvement over the baseline prediction, the RAE (Relative Absolute Error) metric can be used. RAE is calculated as the MAE of the model relative to the MAE of the baseline prediction as defined by Formula 4.5. The RAE columns of Tables 5.1 to 5.4 (highlighted in gray) predominantly display values less than 1, with a few exceptions. This indicates that the models perform better than the baseline prediction for most months. Some of the models, however, show RAE greater than 1 for a few of the months which are highlighted in red in the tables. That means that the baseline prediction outperforms that particular model for those months. It is noteworthy that the RF model has monthly RAE values that are consistently less than 1. This indicates that the RF model outperforms the baseline model for all months.

Uniform vs. weighted

As previously mentioned, each of the tables 5.1 to 5.4 contain the results of two separate models that were built using the two types of CPUE datasets obtained from *uniform* or *weighted* catch distributions, as described in Section 3.2.1. Of course, evaluation of the models are done against the corresponding type of CPUE dataset.

To recapitulate, the need for choosing a distribution method arises from the fact that the catch amount for each fishing trip is only measured at the time of landing; which provides us with the *total* catch for the *whole* trip. Total catch is then

distributed over the fishing segments of the vessel trajectory as described in Section 3.3.5. The choice of distribution method determines how closely the distributed catches simulate the reality.

Even though it is imperative that the catch distribution method simulates the reality of the fishing process as closely as possible, it is not possible to assess the impact of the choice of distribution method based on the evaluation metrics for prediction modelling. The reason is simply because the prediction models are trained and evaluated on CPUE datasets that were created based on the same catch distribution method—in this case either *uniform* or *weighted*. Therefore, comparing the prediction models that were built using the two different distribution methods would not be informative about which distribution method more accurately simulates the true catch distribution.

Best performing model

Table 5.5 provides an overview of the annually averaged evaluation metrics for monthly CPUE predictions for the four machine learning methods. The information is the same as the 13^{th} row (labeled 'All') from the Tables 5.1 to 5.4. RF has the lowest RAE for both *uniform* and *weighted* models. With an annually averaged RAE of 0.87 for weighted and uniform models, RF provides a 13% improvement over the baseline predictions. As mentioned previously, RF is also the only method that exhibited monthly RAE values that were consistently less than 1 for all months, for both *uniform* and *weighted* distributions (Table 5.1). Consequently, RF is the best performing model in our experiment.

Table 5.5. Model comparison (annual metrics)

	Method	MAE	RAE
Uniform catch distribution	RF	1490.18	0.87
	XGBoost	1521.52	0.89
	SVM	1518.04	0.89
	GLM	1533.19	0.90
Weighted catch distribution	RF	1116.69	0.87
	XGBoost	1148.23	0.89
	SVM	1159.26	0.90
	GLM	1170.29	0.91

Summary of the results

All four models produced similar results, with 9 to 13% improvement for monthly CPUE prediction compared to the baseline. The significant model error with respect to the baseline is due to paucity of the data which will be discussed in Section 6.1.1. Nonetheless, Random Forest (RF) performed slightly better than rest of the methods with 13% improvement compared to the baseline for both uniform and weighted models; and by having RAE < 1 for all months.

As an example, Figure 5.1 shows the baseline (left), actual (middle), and predicted monthly CPUE for January 2016 using RF (right). The portrayed CPUE maps pertain to the uniform catch distribution for gear *rapido* and the species *Sardina pilchardus*. Since the fished area in January of 2015 (left) is smaller than the area covered in 2016 (middle), the baseline forecast is missing a number of cells. This limitation is overcome in the RF prediction (right), where we can produce predicted values for any given cell. Even though the RF prediction is an improvement over the baseline, it is evident from Figure 5.1 that the RF model is under-predicting on cells with no fishing activity in January 2015. This issue would likely be mitigated if more historical data were available.

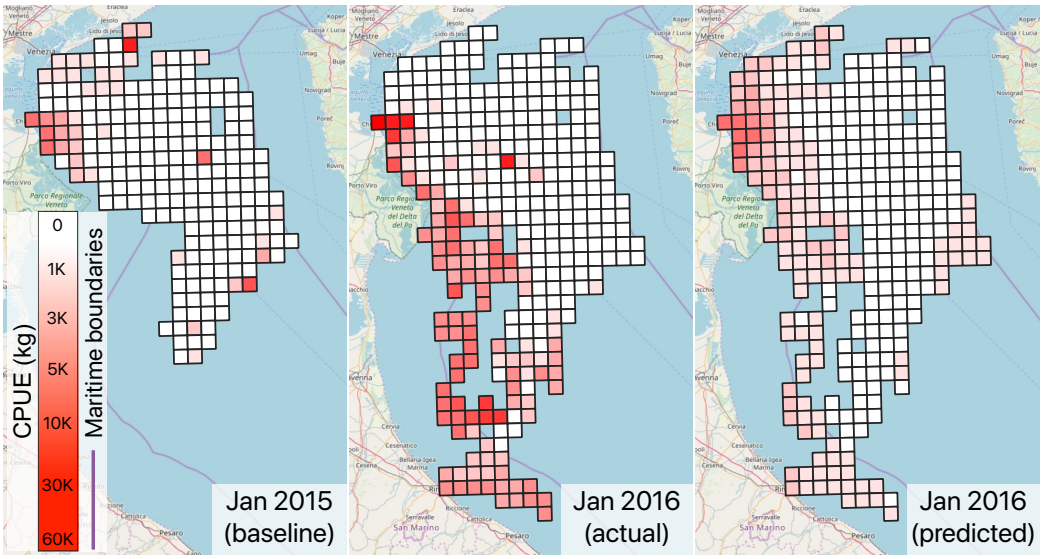

Figure 5.1. January CPUE maps pertaining to the *uniform catch distribution* for gear *Rapido* and the species *Sardina pilchardus*. Actual values for Jan. 2015 (left) are the baseline for Jan. 2016. Actual values for Jan. 2016 (middle) are used in the evaluation of the Random Forests model predictions for Jan. 2016 (right).

Chapter 6

Conclusions and Future Work

In this work, we developed a framework for CPUE prediction using three data sources from the North Adriatic region: (i) AIS vessel tracking data, (ii) daily landing reports dataset, and (iii) related environmental variables. Although CPUE prediction modeling was the final goal of this study, a significant part of this work is represented by the process of transformation and integration of the three heterogeneous data sources to obtain a dataset conducive to the task of prediction modeling. The data modeling and integration process resulted in two different representations of the available data: (i) a set of semantic trajectories (described in Section 3.3), and (ii) gridded[1] spatio-temporal maps of aggregated catches, fishing effort, and CPUE further augmented with environmental factors (described in Section 3.2). The latter representation was used for prediction modeling, but both representations on their own can provide valuable insight into the spatio-temporal aspects of fishing activities, as described below.

By using semantic trajectory modeling and integrating the AIS and landing datasets, an enriched dataset of semantic trajectories was created, a summary of which is given in Section 3.3.6. The trajectories dataset consists of records of distinct fishing trips, each enriched with semantic information such as the vessel's activity and the amount of catch per species for individual segments of the trajectories. This representation allows for asking high level queries about the vessel trajectories, such as "*where was the vessel engaged in fishing activity on a particular trajectory?*", or "*what is the distribution of catch over a particular fishing trip?*".

Subsequently, using the set of semantic trajectories and a spatial grid[1], spatio-temporal maps of aggregate catches, fishing effort, and CPUE were constructed and augmented with environmental factors. These maps can be customized for any choice of spatial and temporal granularity, fishing gear, and species caught; allowing for the

[1]The spatial grid is described in Section 3.2.1.

creation of tailored distribution maps of the aggregate catches, fishing effort, and CPUE. This powerful and flexible representation, achieved through the amalgamation of heterogeneous data sources, is especially valuable to fisheries management by facilitating the spatio-temporal analysis of the aggregate catches, fishing effort, and CPUE per gear and species.

The resulting spatio-temporal CPUE datasets were then used in the CPUE prediction modelling. The prediction task, as specified in Section 4.1, was to predict *monthly* CPUE values for individual spatial cells[1] for 2016, given a *daily* spatio-temporal dataset of CPUE augmented with environmental factors for 2015. Prediction results using the Machine Learning methods, specified in Section 5.1.1, showed improvements between 9 to 13% compared to the baseline prediction; with Random Forests performing slightly better than the other methods. The modest but consistent improvements in the prediction results using Machine Learning methods are suggestive of the potential of such methods for this task. However, the results are also indicative of the limitations of the conducted experiments, and point to the possibility of further improvements, both of which subjects are discussed in the following sections.

6.1 Limitations

This section discusses the limitations that we recognized in this work. We then proceed to propose ideas to address these limitations and expand this work in Section 6.2.

6.1.1 Paucity of the data

The main limitation of this study is the short temporal horizon of the available data. Having a dataset with a short time span of only two years, and having to reserve a part of it for model evaluation, leaves us with a training dataset that is very short for prediction modeling relating to fishing activities—which have seasonal and annual trends by nature. Furthermore, due to the seasonal nature of the data, and because we wanted to evaluate our monthly prediction results on all months of the year, the data for the whole year of 2016 was set aside for evaluation. That left us with only one year of data to train the models on, which is not enough to capture the long term trends. Additionally, paucity of the data limited this study's focus to the analysis of

the gear and species that constitute the largest share of the datasets (i.e. gear *rapido* and species *Sardina pilchardus*).

Sparsity of the data is aggravated by the addition of the spatial dimensions in our approach.[2] As mentioned in Section 2.1.1, in time-series approaches without spatial dimensions, a single CPUE value is calculated for the whole large study area. That guarantees having a CPUE value for all the points in time when fishing activity has occurred *anywhere* in the study area. In contrast, because CPUE values in our dataset are calculated for individual spatial cells, not all the cells have CPUE values for the all the points in time for a period, as evident in Figure 5.1. This means for most cells the set of available CPUE values in time is even smaller than the already limited temporal horizon of the dataset.

As more data becomes available[3] these problems could be mitigated. Also having more data would allow for prediction techniques that take advantage of the autocorrelation of variables in space and time. Such techniques, which will be discussed in Section 6.2, would in turn alleviate the added sparsity of data by taking advantage of the dependencies in the spatial and temporal domains, since they can use information from nearby cells in the spatial and temporal vicinity.

6.1.2 Implicit assumption of spatio-temporal independence

To set the context, *lagged dependency* is referred to the dependency of the response variable on the values of itself or another variable *from an earlier point in time*. When previous values of a variable is used as input to a model, the variable is referred to as a *lagged variable*; and the period it goes back in time is called its *lag*.

In reality, the response variable in our prediction task—i.e., CPUE—could depend on its lagged values in time, and its neighbouring values in space—i.e., spatio-temporal autocorrelation. By disregarding the autocorrelations in our modeling approach, it is implicitly assumed the response variable is independent of its values in the past or in its spatial vicinity; which potentially limits the predictive power of the model.

Furthermore, there could be correlations between CPUE and the lagged values of *other* variables. For example, chlorophyll-a (chl-a) concentration—being an index

[2] As noted before, the ability to combine detailed vessel tracking and landing data made the spatio-temporal approach in this work possible, which also differentiates this study from others.

[3] At the time of writing this document more data has become available. See Section 6.2.

for phytoplankton biomass which is at the base of the aquatic food chain [46]—could have a delayed effect on the amount of catch for different species.

However, the lag between correlated values could be too long to be verified within the limited temporal horizon of our current dataset. For example, a year worth of data might not be enough to verify the presence of seasonally lagged correlations. In fact, a preliminary attempt was made to analyse the spatio-temporal autocorrelation of CPUE which did not reveal strong autocorrelation, possibly due to limited temporal horizon of the data.[4] In Section 6.2 we discuss prediction methods that take these correlations into account.

6.1.3 Rudimentary distribution method for catch

As described in Section 3.3.5, we used two methods to distribute the catch that each vessel reported at the time of landing over the fishing segments of its corresponding trajectory: (i) uniform and (ii) weighted distributions. In the uniform approach, the catch is distributed, over the fishing segments of the trajectory, proportional to the length of each segment. In the weighted approach, the catch is distributed proportional to the length of the fishing segments multiplied by a weight; where the weight is proportional to the count of vessels that were fishing during that *day* in the same grid cell that the trajectory segment falls into. The weighted approach aims to improve the uniform approach in terms of simulating the true catch distribution, based on the idea that areas with more fishing activities are likely to have higher catch rates. We recognize that both approaches are limited in their ability to simulate the real catch distribution, and have proposed suggestions to improve the distribution in Section 6.2.

6.1.4 Temporal granularity mismatch

As noted in Section 4.1.2, to obtain *monthly* predictions for CPUE while considering the *daily* effects of environmental factors (e.g., wave height) on the fishing activities, we took an indirect approach. Specifically, first *daily* CPUE values were predicted for each grid cell, and then the values were averaged over the month to obtain the

[4]The autocorrelation analysis of CPUE was performed using the variogramST function from the gstat R package [60].

monthly prediction for the cell. This indirect approach may limit the predictive power compared to direct prediction for the month due to the following reasons. First off, calculating CPUE values for shorter time periods results in higher variance—similar to the effect of higher spatial granularity, i.e., smaller cell size—which could result in less accurate predictions. Furthermore, some environmental factors might have a more significant delayed and/or cumulative effects, rather than effecting CPUE on the same day as they were measured. In particular, as noted in Section 6.1.2, chl-a concentration could have a delayed effect on the amount catch. Besides, a direct prediction approach would result in a simpler forecasting pipeline than the indirect approach; and reducing the complexity of the pipeline is generally desirable. In Section 6.2 we will propose alternative approaches to address this issue.

6.2 Future work

At the time of writing this document, the data for 2017 and 2018 has also become available; which extends the current dataset used in this study to cover the total of four years between 2015 and 2018. As more data becomes available, this work can be expanded by considering a wider range of modelling techniques for the predictive analysis. In this section, we discuss a few of such techniques, suggest ideas to address the limitations mentioned in the previous section, and also propose other possible applications of the rich set of data sources used in this study.

6.2.1 Improving the predictive analysis

As noted in Section 6.1.2, by taking advantage of the correlation of CPUE values with their lagged values in time or their nearby values in the spatial vicinity, the predictive power of the forecast models can be improved. This type of correlation is considered spatio-temporal autocorrelation, and relies on the assumption of continuity of the variable in space and time. As mentioned in Section 2.1.1, moving average autoregressive models for time-series data (without spatial dimensions), such as ARIMA, are prevalent in time-series forecasting. STARMA (Space Time AutoRegressive Moving Average) model follows the same principles, but is extended for space-time modelling based on [62]. Alternatively, spTimer is based on hierarchical Bayesian modelling of point-referenced space-time data, which is available as an R package [3]. Both

STARMA and spTimer are methods that take spatio-temporal autocorrelation into account, and can be used for spatio-temporal CPUE forecasting. However, both of the mentioned methods only work for univariate datasets, and cannot handle independent explanatory variables such as the environmental variables used in this study. On the other hand, spBayes is a method that can fit both univariate and multivariate spatio-temporal models using Markov chain Monte Carlo (MCMC), and is available an R package [30]. As a non-statistical approach, RNN (Recurrent Neural Network) based spatio-temporal prediction models have been used for problems such as decease prediction, traffic forecasting, meteorology, and oceanography [79]. Employing prediction techniques with spatio-temporal 'memory', such as the techniques mentioned above, can potentially improve the spatio-temporal forecast of CPUE.

Some environmental variables (e.g., chl-a concentration) can also have a delayed effect on the CPUE values, as noted in Section 6.1.2. Unlike autocorrelation, this type of lagged correlation is between the response variable (i.e., CPUE) and the independent variables (i.e., environmental variables). Such correlations can be examined by performing time-lagged correlation analysis between CPUE and the environmental variables. For example, [47] investigates the time-lagged response of anchovy CPUE to different environmental variables including chl-a concentration and sea surface temperature (sst); and proceeds to use the time lags which show the highest correlations for modelling. A similar approach can be used in in the context of this study to improve the prediction results.

Temporal granularity mismatch, as described in Section 6.1.4, points out the issues with the indirect approach of obtaining *monthly* CPUE predictions from *daily* predicted values. Using feature engineering, we can adapt the features to be used for direct monthly prediction modelling. However, any alteration of the features needs to be considered carefully, to preserve as much information as possible regarding the daily cause and effect of the features on CPUE—which was the reason for using the indirect approach in this study as noted in Section 4.1.2. Some environmental variables, such as chl-a concentration and sea surface temperature (sst), are likely to be effective for prediction modelling even if averaged monthly. For example, the environmental variables used in the study mentioned in the previous paragraph ([47]) were *monthly* averages. However, other environmental variables are likely to have a more

pronounced effect on the catch amounts on a *daily* basis. For example, turbulent water—quantified by wave height—has an immediate effect on fishing activities; and its average over the month loses that information. Transforming such variables to be used for direct monthly prediction modeling, while preserving the daily cause and effect information, can be done by careful feature engineering informed by reasoning, and expert knowledge. For instance, in the case of wave height, we propose the following procedure to transform the feature to be used in direct monthly prediction modeling, without losing its daily effect on CPUE. First, classify wave height—which is a continuous value measured in meters—into a set of categories based on the wave severity. For example, the Douglas Sea Scale classifies waves into nine categories with descriptive names such as *calm, moderate, rough*, etc [56]. A similar classification can be adopted for our application based on expert knowledge about how wave height affects fishing activities. Using the wave height classification, a monthly count of days for each category of wave height can be obtained; e.g., *calm: 12 days, rough: 10 days*, etc. Then, the monthly counts for the wave height categories can be used as input features for prediction modelling. This approach ensures that the daily effect of wave height on fishing activities is not dismissed, while allowing for direct monthly prediction modelling.

Finally, the availability of more data opens doors to further improve and expand on the predictive analysis. For example, the baseline prediction (described in Section 4.2) can be improved by using more historical data instead of only using the data for 2015. Particularly, the issue of unavailable CPUE values for numerous grid cells in the baseline prediction, as evident in Figure 5.1, can be mitigated if we use the average of historical CPUE values for more years to obtain the baseline predictions. This approach would also have the befit of reducing the variance in the baseline prediction values. Besides, with more data available, it would be feasible to consider analyzing other types of gear and species rather than focusing on the ones that constitute the largest share of the data, as was done in this study.

6.2.2 Improving the catch distribution

The (hypothetical) true distribution of catch along the vessel trajectory would be equivalent to having records of local catch amounts at all points on the vessel's trajectory; which is obviously not possible to attain in a commercial setting. To approximate the catch distribution over the trajectories, two distribution methods were introduced in Section 3.3.5; i.e., *uniform* and *weighted*. The latter method aimed to improve the former based on the idea that areas with a higher count of vessels engaged in fishing activities in a 24 hour window are likely to have higher catch rates on that day. This method can be further improved by experimenting with longer time windows (e.g., weekly), or perhaps considering a moving average of vessel counts with the longer time window. The longer time window would reduce the variance in vessel counts, which could provide a better measure for sustained fishing activities in the area. Alternatively, more sophisticated distribution methods can be utilized that employ existing regional surveys of species abundance to produce more accurate approximations of the catch distributions [44].

6.2.3 Alternative applications

Data modelling and integration of the rich data sources available in this study resulted in two high-level representations of the combined data: (i) the set of semantic trajectories of the fishing trips (described in Section 3.3), and (ii) the detailed spatio-temporal maps of catch, fishing effort, and CPUE per fishing gear and species (described in Section 3.2). Although the second representation was used for prediction modelling in this study, both representations can be the basis for other applications in fisheries management. For example, the high-resolution spatio-temporal maps of fishing effort and CPUE can be used in the analysis of the effects of policies that restrict fishing activities in the North Adriatic sea. Paper [66] uses spatio-temporal maps of fishing effort to evaluate the simulated effects of spatial and temporal closures in the northern and central Adriatic sea to rebuild the stock of a single species (common sole). The landing dataset used in our study allows for generating spatio-temporal maps of fishing effort for several different species, which would allow studying the effects of closure policies on a wider variety of species. Also, the spatio-temporal CPUE datasets can be used in a similar type of analysis.

As mentioned in Section 3.1.2, the landing dataset in this study includes daily catch amounts of 104 different species. Detailed information about the species of catch opens the possibility of analysing non-selective exploitation patterns in the north Adriatic sea. This refers to the problem of catching unintended species while fishing for certain target species. The species that were caught unintentionally are referred to as *bycatch*. Considering that bycatch problem in the trawl fishery—which is the source of the catch data in this study—is seriously harming the sustainability of the fishing activities in the Adriatic region [29], analysing non-selective exploitation patterns is critical.

Appendix A

Model Hyperparameters

As noted in Section 5.1.1, model hyperparameters were selected by performing a grid search for each model, and hyperparameter values that resulted in the lowest mean absolute error on the validation dataset were selected. Here we discuss the models hyperparameters that were selected via grid search, and their selected values are reported in Tables A.2, A.3, A.4, and A.5 for GLM, SVM, XGboost, and Random Forests respectively. Two grid searches were performed for each method, one to build a model on the *uniform* CPUE dataset and one for the *weighted* dataset. Best hyperparameters are reported for both models in each table.

A.1 Generalized linear model (GLM)

As noted in Section 5.1.1, we used H2O's [37] implementation of GLM. H2O's GLM supports elastic net regularization, which is a combination of L1 (Lasso regression) and L2 (ridge regression) regularization methods [55]. Elastic net regularization is parameterized by two hyperparameters: `alpha` and `lambda`, which are described below. H2O GLM also supports Tweedie distributions, which are a family of distributions that consist of normal, Poisson, gamma distributions, and their combinations [55]. Parameterization of the Tweedie distribution in H2O is done by the `tweedie_variance_power` hyperparameter, which is described below.

- `alpha` specifies the distribution between L1 and L2 norms in the elastic net penalty [55]. It can take any value between 0 and 1, where `alpha`=0 is equivalent to L2 (ridge regression), `alpha`=1 would be equivalent to L1 (Lasso regression), and values in between would mean a combination of both.

- `lambda` is a strictly positive number that specifies the penalty strength [55]. As a part of the GLM algorithm, H2O performs a `lambda` search over a range of possible values that are determined heuristically [36]. Therefore, `lambda`

parameter was not chosen via grid search, but the `lambda` values reported in Table A.2 are chosen by H2O's heuristic.

- `tweedie_variance_power` (p) parameterizes the Tweedie distribution. p can take all values except in the interval $(0, 1)$. Some p values result in special cases of Tweedie distribution [55], which are listed in Table A.1.

Table A.1. Tweedie variance power (p) special cases

Tweedie variance power (p)	Distribution
$p = 0$	Normal
$p = 1$	Poisson
$p \in (1, 2)$	Compound Poisson
$p = 2$	Gamma
$p = 3$	Inverse-Gaussian

Table A.2. GLM hyperparamters

Parameter	Best value		Searched values
	uniform	weighted	
`alpha`	1	1	0, 0.05, ..., 0.95, 1.0
`lambda`	2.162e-6	5.953e-6	H2O heuristic†
`tweedie_variance_power`	2.9	2.8	0, 1, 1.1, ..., 2.9, 3, 5, 7

† `lambda` value chosen automatically by H2O from a range of values determined heuristically [36].

A.2 SVM

As noted in Section 5.1.1, we used Scikit-learn's [61] implementation of SVM, which uses the LIBSVM library [13]. Hyperparamters that are selected via grid search are as follows.

- `kernel` specifies the type of kernel to be used with the SVM algorithm. As mentioned in Section 5.1.1, SVM utilizes kernel functions. Scikit-learn [61] provide four pre-computed kernels to choose from — linear, polynomial, radial basis function (rbf), and sigmoid. All four kernels were considered through the grid search.

- `C` is a strictly positive number that specifies the regularization parameter. `C` is inversely proportional to the strength of regularization [13]. Higher values of `C` would result in higher model specificity by forcing smaller hyper-plane margins, and vice versa.

- `epsilon` is a strictly positive number that specifies the epsilon tube. That is, if the distance between a predicated and actual training point is within the epsilon threshold, the loss function does not associate a penalty with that sample point [13]. Smaller epsilon values would result in higher model specificity, and vice versa.

Table A.3. SVM hyperparamters

Parameter	Best value		Searched values
	uniform	weighted	
`kernel`	poly (cubic)	poly (cubic)	linear, poly (cubic), rbf, sigmoid
`C`	1000	1000	0.001, 0.01, 0.1, 1, 10, 100 , 1000
`epsilon`	0.1	0.001	0, 0.001, 0.01, 0.1, 1

A.3 XGBoost

As noted in Section 5.1.1, we used H2O's [37] implementation of XGBoost. Hyper-paramters that are selected via grid search are as follows.

- `ntrees` specifies the total number of trees to build as base learners for the tree boosting algorithm; which also would be equal to the number of boosting iterations. Although higher number of trees would result in overfitting, H2O [37] uses an early stopping criterion based on evaluation on the validation dataset to keep the number of trees as low as possible. XGBoost also uses additional regularization techniques, such as *'learning rate'* and *'sample rate'*, which are described below.

- `max_depth` specifies the maximum depth of the trees. The deeper the trees, the more complex the model would be, potentially resulting in overfitting [36]. Grid search is used to find the smallest appropriate value for this parameter.

- `min_rows` specifies the minimum number of samples needed to split a node [36]. As the value of this parameter increases, each tree becomes more constrained as it considers more samples at each node, which could result in underfitting [36].

- `learn_rate` specifies the learning rate, which corresponds to the *shrinkage* parameter in the boosting algorithm [36]. Small learning rates (e.g. less than 0.1) greatly improves model generalization at the expense of increasing training time [38].

- `sample_rate` specifies the sampling ratio of the training data [36]. Values for this parameter range from 0 to 1. Sample rate of 0.5 means that half of the data is randomly sampled without replacement to build each base learner. Smaller values for sample rates introduce randomness into the algorithm and help reduce overfitting [32].

Table A.4. XGBoost hyperparamters

Parameter	Best value		Searched values
	uniform	weighted	
`ntrees`	30	30	20 to 200
`max_depth`	5	5	2 to 50
`min_rows`	7	5	1 to 50
`learn_rate`	0.01	0.01	0.01 to 1
`sample_rate`	0.5	0.5	0.01 to 1

A.4 Random Forests

As noted in Section 5.1.1, we used H2O's [37] implementation of Random Forests. Hyperparamters that are selected via grid search are as follows.

- `ntrees` specifies the total number of trees as base learners in the RF model. In paper [11], it is proven that RF does not overfit as the number of trees increase, but the generalization error converges to a limiting value. However, the training time is proportionally related to the number of trees times the

number of training samples [18]. Therefore, it is desirable for the number of trees to be as small as possible without compromising the model accuracy. To limit the number of trees, H2O monitors the improvement in model goodness-of-fit (regression deviance), and stops adding trees if the model is not improving within a certain threshold.

- `max_depth` specifies the maximum depth of each tree. This parameter together with the number of trees determines the size of the model [18]. Therefore, deeper trees, especially depths greater than 10, increase the computing time required [36]. Deeper trees generally improve the accuracy on the training set and can result in overfitting [36]. A grid search is recommended to find the appropriate value for this parameter.

- `min_rows` specifies the minimum number of samples needed to split a node [36]. As the value of this parameter increases, each tree becomes more constrained as it considers more samples at each node, which could result in underfitting [36].

- `nbins` specifies the number of bins for building the histograms of each *numerical* feature at the node level [36]. Boundaries of the bins are then used as potential split points. As `nbins` increases, the model more closely approximates considering each sample as an split point, resulting in more model specificity (overfitting) and vice versa [36].

- `nbins_cats` specifies the number of bins for building the histograms of each *categorical* feature at the node level, similar to `nbins` [36]. However, an increase `nbins_cats` value has a far greater effect than `nbins` on model generalization [37]. Mainly because larger values for `nbins` would lead to more accurate numerical splits for numerical features; but larger values `nbins_cats` can result in perfect splitting on categorical features, leading to overfitting. Since the only categorical feature of our data set is '*Seasons*' with 4 categories, small values for `nbins_cats` are considered (between 2 and 4).

Table A.5. Random Forests hyperparamters

Best value	Selected value		Searched values
	uniform	weighted	
`ntrees`	41	76	20 to 200
`max_depth`	17	7	1 to 30
`min_rows`	28	40	1 to 50
`nbins`	10	8	5 to 500
`nbins_cats`	4	2	2 to 4

Bibliography

[1] Pedram Adibi, Fabio Pranovi, Alessandra Raffaetà, Elisabetta Russo, Claudio Silvestri, Marta Simeoni, Amilcar Soares, and Stan Matwin. Predicting fishing effort and catch using semantic trajectories and machine learning. In *International Workshop on Multiple-Aspect Analysis of Semantic Trajectories*, pages 83–99. Springer, 2019.

[2] J. Scott Armstrong and Fred Collopy. Error measures for generalizing about forecasting methods: Empirical comparisons. *International Journal of Forecasting*, 8(1):69–80, 1992.

[3] Khandoker Shuvo Bakar and Sujit K. Sahu. spTimer: Spatio-temporal bayesian modeling using R. *Journal of Statistical Software*, 63(15):1–32, 2015.

[4] Hadiza Yakubu Bako, Mohd Saifullah Rusiman, Ibrahim Lawal Kane, and Hazel Monica Matias-Peralta. Predictive modeling of pelagic fish catch in malaysia using seasonal arima models. *Agriculture, Forestry and Fisheries*, 2(3):136–140, 2013.

[5] Heather Ball. *Satellite Ais for Dummies*. John Wiley & Sons Incorporated, 2013.

[6] Scott P Bannerot and C Bruce Austin. Using frequency distributions of catch per unit effort to measure fish-stock abundance. *Transactions of the American Fisheries Society*, 112(5):608–617, 1983.

[7] Francois Bastardie, Silvia Angelini, Luca Bolognini, Federico Fuga, Chiara Manfredi, Michela Martinelli, J Rasmus Nielsen, Alberto Santojanni, Giuseppe Scarcella, and Fabio Grati. Spatial planning for fisheries in the northern adriatic: working toward viable and sustainable fishing. *Ecosphere*, 8(2):e01696, 2017.

[8] Francois Bastardie, J Rasmus Nielsen, Clara Ulrich, Josefine Egekvist, and Henrik Degel. Detailed mapping of fishing effort and landings by coupling fishing logbooks with satellite-recorded vessel geo-location. *Fisheries Research*, 106(1):41–53, 2010.

[9] Vania Bogorny, Chiara Renso, Artur Ribeiro de Aquino, Fernando de Lucca Siqueira, and Luis Otavio Alvares. Constant–a conceptual data model for semantic trajectories of moving objects. *Transactions in GIS*, 18(1):66–88, 2014.

[10] Leo Breiman. Bagging Predictors. *Machine Learning*, 24(2):123–140, August 1996.

[11] Leo Breiman. Random Forests. *Machine Learning*, 45(1):5–32, October 2001.

[12] Piera Carpi, Giuseppe Scarcella, and Massimiliano Cardinale. The saga of the management of fisheries in the adriatic sea: history, flaws, difficulties, and successes toward the application of the common fisheries policy in the mediterranean. *Frontiers in Marine Science*, 4:423, 2017.

[13] Chih-Chung Chang and Chih-Jen Lin. Libsvm: A library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3):1–27, 2011.

[14] Shui-Kai Chang and Tzu-Lun Yuan. Deriving high-resolution spatiotemporal fishing effort of large-scale longline fishery from vessel monitoring system (vms) data and validated by observer data. *Canadian Journal of Fisheries and Aquatic Sciences*, 71(9):1363–1370, 2014.

[15] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

[16] Patricia M. Clay, Ian Cowx, David Evans, Felimon Gayanilo, Richard Grainger, Angel Gumy, Veravat Hongskul, Tony Jarrett, Paul Medley, Peter Miyake, Sean Pascoe, Christian Riise, Per Sparre, Constantine Stamatopoulos, Siebren Venema, Morten Vinther, Teo Wan, and Paul A.M. Zwieten. Guidelines for the routine collection of capture fishery data. *FAO Fisheries Technical Paper*, 382, 01 1998.

[17] Francesco Colloca, Massimiliano Cardinale, Francesc Maynou, Marianna Giannoulaki, Giuseppe Scarcella, Klavdija Jenko, Josè Maria Bellido, and Fabio Fiorentino. Rebuilding mediterranean fisheries: a new paradigm for ecological sustainability. *Fish and fisheries*, 14(1):89–109, 2013.

[18] Darren Cook. *Practical machine learning with H2O: powerful, scalable techniques for deep learning and AI.* " O'Reilly Media, Inc.", 2016.

[19] Copernicus: Europe's eyes on Earth. `https://www.copernicus.eu/en`.

[20] Ivone Alejandra Czerwinski, Juan Carlos Gutiérrez-Estrada, and José Antonio Hernando-Casal. Short-term forecasting of halibut cpue: Linear and non-linear univariate approaches. *Fisheries Research*, 86(2-3):120–128, 2007.

[21] Erico N de Souza, Kristina Boerder, Stan Matwin, and Boris Worm. Improving fishing pattern detection from satellite ais using data mining and machine learning. *PloS one*, 11(7):e0158248, 2016.

[22] DB DeLury. On the planning of experiments for the estimation of fish populations. *Journal of the Fisheries Board of Canada*, 8(4):281–307, 1951.

[23] Roy Deng, Cathy Dichmont, David Milton, Mick Haywood, David Vance, Natasha Hall, and David Die. Can vessel monitoring system data also be used to study trawling intensity and population depletion? the example of australia's northern prawn fishery. *Canadian Journal of Fisheries and Aquatic Sciences*, 62(3):611–622, 2005.

[24] Thomas G. Dietterich. Ensemble Methods in Machine Learning. In *Multiple Classifier Systems*, volume 1857 of *LNCS*, pages 1–15. Springer, 2000.

[25] TA Dinmore, DE Duplisea, BD Rackham, DL Maxwell, and S Jennings. Impact of a large-scale area closure on patterns of fishing disturbance and the consequences for benthic communities. *ICES Journal of Marine Science*, 60(2):371–380, 2003.

[26] A Dunn, SJ Harley, IJ Doonan, and B Bull. Calculation and interpretation of catch-per-uniteffort (cpue) indices. *New Zealand fisheries assessment report*, 1:44, 2000.

[27] Torkild Eriksen, Gudrun Høye, Bjørn Narheim, and Bente Jensløkken Meland. Maritime traffic monitoring using a space-based ais receiver. *Acta Astronautica*, 58(10):537–549, 2006.

[28] Amending directive 2002/59/ec of the european parliament and of the council establishing a community vessel traffic monitoring and information system. `http://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32011L0015&from=EN`. Accessed: 2019-11-13.

[29] GFCM FAO. The state of mediterranean and black sea fisheries. *Rome: FAO*, 2016.

[30] Andrew O Finley, Sudipto Banerjee, and Bradley P Carlin. spbayes: an r package for univariate and multivariate hierarchical point-referenced spatial models. *Journal of statistical software*, 19(4):1, 2007.

[31] Jerome Friedman, Trevor Hastie, Robert Tibshirani, et al. Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). *The annals of statistics*, 28(2):337–407, 2000.

[32] Jerome H Friedman. Stochastic gradient boosting. *Computational statistics & data analysis*, 38(4):367–378, 2002.

[33] Hans Gerritsen and Colm Lordan. Integrating vessel monitoring systems (vms) data with daily catch data from logbooks to explore the spatial distribution of catch and effort at high resolution. *ICES Journal of Marine Science*, 68(1):245–252, 2010.

[34] Pauline Gloaguen, Stéphanie Mahévas, Etienne Rivot, Matthieu Woillez, Jérôme Guitton, Youen Vermard, and Marie-Pierre Etienne. An autoregressive model to describe fishing vessel movement and activity. *Environmetrics*, 26(1):17–28, 2015.

[35] C Phillip Goodyear. Spatio-temporal distribution of longline catch per unit effort, sea surface temperature and atlantic marlin. *Marine and Freshwater Research*, 54(4):409–417, 2003.

[36] H2O.ai. H2O Docs - Appendix A - Parameters. `http://docs.h2o.ai/h2o/latest-stable/h2o-docs/parameters.html`. Accessed: 2020-2-23.

[37] H2O.ai. H2O, Feb. 2020. Version 3.28.0.2. `https://github.com/h2oai/h2o-3`.

[38] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The elements of statistical learning: data mining, inference, and prediction*. Springer Science & Business Media, 2009.

[39] MR Heath. Field investigations of the early life stages of marine fish. In *Advances in marine biology*, volume 28, pages 1–174. Elsevier, 1992.

[40] Rob J. Hyndman and George Athanasopoulos. *Forecasting: principles and practice*. OTexts, 2 edition, 2018.

[41] Regulations for carriage of AIS. `http://www.imo.org/en/OurWork/safety/navigation/pages/ais.aspx`. Accessed: 2019-11-13.

[42] International Maritime Organization (IMO). *Revised Guidelines For The Onboard Operational Use Of Shipborne Automatic Identification Systems (AIS)*, December 2015.

[43] Edgar Lanz, Manuel Nevarez-Martinez, Juana López-Martínez, and JUAN A Dworak. Small pelagic fish catches in the gulf of california associated with sea surface temperature and chlorophyll. *CalCOFI Rep*, 50:134–146, 2009.

[44] Ethan Lawler and Joanna Mills Flemming. personal communication, Nov. 2019.

[45] Janette Lee, Andy B South, and Simon Jennings. Developing reliable, repeatable, and accessible methods to provide high-resolution estimates of fishing-effort distributions from vessel monitoring system (vms) data. *ICES Journal of Marine Science*, 67(6):1260–1271, 2010.

[46] Rebecca Lindsey, Michon Scott, and R Simmon. What are phytoplankton. *NASA's Earth Observatory. Available on http://earthobservatory. nasa. gov/Librar y/phytoplankton*, 2010.

[47] Paloma Martín, Nixon Bahamon, Ana Sabatés, Francesc Maynou, Pilar Sánchez, and Montserrat Demestre. European anchovy (engraulis encrasicolus) landings and environmental conditions on the catalan coast (nw mediterranean) during 2000–2005. In *Essential Fish Habitat Mapping in the Mediterranean*, pages 185–199. Springer, 2008.

[48] Shannon M McCluskey and Rebecca L Lewison. Quantifying fishing effort: a synthesis of current methods and their applications. *Fish and fisheries*, 9(2):188–200, 2008.

[49] Ronaldo dos Santos Mello, Vania Bogorny, Luis Otavio Alvares, Luiz Henrique Zambom Santana, Carlos Andres Ferrero, Angelo Augusto Frozza, Geomar Andre Schreiner, and Chiara Renso. MASTER: A multiple aspect view on trajectories. *Transactions in GIS*, 2019. to appear.

[50] Craig M Mills, Sunny E Townsend, Simon Jennings, Paul D Eastwood, and Carla A Houghton. Estimating high resolution trawl fishing effort from satellite-based vessel monitoring system data. *ICES Journal of Marine Science*, 64(2):248–255, 2006.

[51] Liz Morris and David Ball. Habitat suitability modelling of economically important fish species with commercial fisheries data. *ICES Journal of Marine Science*, 63(9):1590–1603, 2006.

[52] David Mountain and Jonathan Raper. Modelling human spatio-temporal behaviour: a challenge for location-based services. In *Proceedings of 6th International Conference on Geocomputation*, 2001.

[53] Steven A Murawski, Susan E Wigley, Michael J Fogarty, Paul J Rago, and David G Mountain. Effort distribution and catch patterns adjacent to temperate mpas. *ICES Journal of Marine Science*, 62(6):1150–1167, 2005.

[54] Fabrizio Natale, Maurizio Gibin, Alfredo Alessandrini, Michele Vespe, and Anton Paulrud. Mapping fishing effort through ais data. *PloS one*, 10(6):e0130746, 2015.

[55] Tomas Nykodym, Tom Kraljevic, Nadine Hussami, Ariel Rao, and Amy Wang. Generalized linear modeling with H2O. February 2020: Seventh Edition.

[56] Edward H. Owens. Sea conditions. In *Beaches and Coastal Geology*, pages 722–722. Springer US, Boston, MA, 1984.

[57] Christine Parent, Stefano Spaccapietra, Chiara Renso, Gennady Andrienko, Natalia Andrienko, Vania Bogorny, Maria Luisa Damiani, Aris Gkoulalas-Divanis, Jose Macedo, Nikos Pelekis, et al. Semantic trajectories modeling and analysis. *ACM Computing Surveys (CSUR)*, 45(4):42, 2013.

[58] DG Parsons and EB Colbourne. Forecasting fishery performance for northern shrimp (pandalus borealis) on the labrador shelf (nafo divisions 2hj). *Journal of Northwest Atlantic Fishery Science*, 27:11–20, 2000.

[59] RK Paul, MK Das, et al. Statistical modelling of inland fish production in india. *Journal of the Inland Fisheries Society of India*, 42(2):1–7, 2010.

[60] Edzer J. Pebesma. Multivariable geostatistics in S: the gstat package. *Computers & Geosciences*, 30:683–691, 2004.

[61] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[62] Phillip E Pfeifer and Stuart Jay Deutrch. A three-stage iterative procedure for space-time modeling phillip. *Technometrics*, 22(1):35–47, 1980.

[63] JT Reubens, Ulrike Braeckman, Jan Vanaverbeke, Carl Van Colen, Steven Degraer, and Magda Vincx. Aggregation at windmill artificial reefs: Cpue of atlantic cod (gadus morhua) and pouting (trisopterus luscus) at different habitats in the belgian part of the north sea. *Fisheries Research*, 139:28–34, 2013.

[64] AD Rijnsdorp, AM Buys, F Storbeck, and EG Visser. Micro-scale distribution of beam trawl effort in the southern north sea between 1993 and 1996 in relation to the trawling frequency of the sea bed and the impact on benthic organisms. *ICES Journal of Marine Science*, 55(3):403–419, 1998.

[65] T Russo, A Parisi, and S Cataudella. Spatial indicators of fishing pressure: Preliminary analyses and possible developments. *Ecological indicators*, 26:141–153, 2013.

[66] Giuseppe Scarcella, Fabio Grati, Saša Raicevich, Tommaso Russo, Roberto Gramolini, Robert D Scott, Piero Polidori, Filippo Domenichetti, Luca Bolognini, Otello Giovanardi, et al. Common sole in the northern and central adriatic sea: spatial management scenarios to rebuild the stock. *Journal of sea research*, 89:12–22, 2014.

[67] MP Lincoln Smith, JD Bell, DA Pollard, and BC Russell. Catch and effort of competition spearfishermen in southeastern australia. *Fisheries Research*, 8(1):45–61, 1989.

[68] Amílcar Soares Júnior, Bruno Neiva Moreno, Valéria Cesário Times, Stan Matwin, and Lucídio dos Anjos Formiga Cabral. Grasp-uts: an algorithm for unsupervised trajectory segmentation. *International Journal of Geographical Information Science*, 29(1):46–68, 2015.

[69] HU Solanlki, RM Dwivedi, SR Nayak, JV Jadeja, DB Thakar, HB Dave, and MI Patel. Application of ocean colour monitor chlorophyll and avhrr sst for fishery forecast: Preliminary validation results off gujarat coast, northwest coast of india. *Indian Journal of Marine Sciences*, 30:132–138, 2001.

[70] Stefano Spaccapietra and Christine Parent. Adding meaning to your steps (keynote paper). In *International Conference on Conceptual Modeling*, pages 13–31. Springer, 2011.

[71] Stefano Spaccapietra, Christine Parent, Maria Luisa Damiani, Jose Antonio de Macedo, Fabio Porto, and Christelle Vangenot. A conceptual view on trajectories. *Data & knowledge engineering*, 65(1):126–146, 2008.

[72] KI Stergiou. Short-term fisheries forecasting: comparison of smoothing, arima and regression techniques. *Journal of Applied Ichthyology*, 7(4):193–204, 1991.

[73] Chen-Te Tseng, Chi-Lu Sun, Su-Zan Yeh, Shih-Chin Chen, and Wei-Cheng Su. Spatio-temporal distributions of tuna species and potential habitats in the western and central pacific ocean derived from multi-satellite data. *International Journal of Remote Sensing*, 31(17-18):4543–4558, 2010.

[74] Efthymia V Tsitsika, Christos D Maravelias, and John Haralabous. Modeling and forecasting pelagic fish production using univariate and multivariate arima models. *Fisheries science*, 73(5):979–988, 2007.

[75] Vladimir Vapnik, Steven E Golowich, and Alex J Smola. Support vector method for function approximation, regression estimation and signal processing. In *Advances in neural information processing systems*, pages 281–287, 1997.

[76] Paraskevas Vasilakopoulos, Christos D Maravelias, and George Tserpes. The alarming decline of mediterranean fish stocks. *Current Biology*, 24(14):1643–1648, 2014.

[77] William T Vickers. Hunting yields and game composition over ten years in an amazon indian territory. *Neotropical wildlife use and conservation*, 400:53–81, 1991.

[78] Jilin Zhang, Jiali Geng, Jian Wan, Yifan Zhang, Mingwei Li, Jue Wang, and Neal N Xiong. An automatically learning and discovering human fishing behaviors scheme for cpscn. *IEEE Access*, 6:19844–19858, 2018.

[79] Ali Ziat, Edouard Delasalles, Ludovic Denoyer, and Patrick Gallinari. Spatio-temporal neural networks for space-time series forecasting and relations discovery. In *2017 IEEE International Conference on Data Mining (ICDM)*, pages 705–714. IEEE, 2017.