2017-06

# Predicting vessel trajectories from AIS data using R

Young, Brian L.

Monterey, California: Naval Postgraduate School


# NAVAL POSTGRADUATE SCHOOL

**MONTEREY, CALIFORNIA**

# THESIS

**PREDICTING VESSEL TRAJECTORIES FROM AIS DATA USING R**

by

Brian L. Young

June 2017

Thesis Advisor: Robert A. Koyak
Second Reader: Samuel H. Huddleston

THIS PAGE INTENTIONALLY LEFT BLANK

| REPORT DOCUMENTATION PAGE | | | *Form Approved OMB No. 0704-0188* |
|---|---|---|---|



| **1. AGENCY USE ONLY** *(Leave blank)* | **2. REPORT DATE** June 2017 | **3. REPORT TYPE AND DATES COVERED** Master's thesis |
|---|---|---|

| **4. TITLE AND SUBTITLE** PREDICTING VESSEL TRAJECTORIES FROM AIS DATA USING R | **5. FUNDING NUMBERS** |
|---|---|
| **6. AUTHOR(S)** Brian L. Young | |
| **7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)** Naval Postgraduate School Monterey, CA 93943-5000 | **8. PERFORMING ORGANIZATION REPORT NUMBER** |
| **9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES)** N/A | **10. SPONSORING / MONITORING AGENCY REPORT NUMBER** |

**11. SUPPLEMENTARY NOTES** The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB number ____N/A____.

| **12a. DISTRIBUTION / AVAILABILITY STATEMENT** Approved for public release. Distribution is unlimited. | **12b. DISTRIBUTION CODE** |
|---|---|

**13. ABSTRACT (maximum 200 words)**

Analysts and security experts seek automated algorithms to predict future behavior of vessels at sea based on Automated Identification System (AIS) data. This thesis seeks to accurately predict the future location of a vessel at sea based on cluster analysis of historical vessel trajectories using a random forest. Once similar trajectories have been clustered into a route, expected prediction error can be empirically estimated based on an independent validation data set not used during training, then applied to an independent test set to produce an expected prediction region with a user-defined level of expectation. Our results show that the prediction region contains the true interpolated future position at the expectation level set by the user, therefore producing a valid methodology for both estimating the future vessel location and for assessing anomalous vessel behavior.

| **14. SUBJECT TERMS** AIS, predict, route, trajectory, cluster, neural network, model, random forest | **15. NUMBER OF PAGES** 75 |
|---|---|
| | **16. PRICE CODE** |

| **17. SECURITY CLASSIFICATION OF REPORT** | **18. SECURITY CLASSIFICATION OF THIS PAGE** | **19. SECURITY CLASSIFICATION OF ABSTRACT** | **20. LIMITATION OF ABSTRACT** |
|---|---|---|---|
| Unclassified | Unclassified | Unclassified | UU |

NSN 7540-01-280-5500 Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

**Approved for public release. Distribution is unlimited.**

**PREDICTING VESSEL TRAJECTORIES FROM AIS DATA USING R**

Brian L. Young
Major, United States Army Reserve
B.S., Oklahoma State University, 1999

Submitted in partial fulfillment of the
requirements for the degree of

**MASTER OF SCIENCE IN OPERATIONS RESEARCH**

from the

**NAVAL POSTGRADUATE SCHOOL**
**June 2017**

Approved by: Robert A. Koyak
Thesis Advisor

Samuel H. Huddleston
Second Reader

Patricia Jacobs
Chair, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

# ABSTRACT

Analysts and security experts seek automated algorithms to predict future behavior of vessels at sea based on Automated Identification System (AIS) data. This thesis seeks to accurately predict the future location of a vessel at sea based on cluster analysis of historical vessel trajectories using a random forest. Once similar trajectories have been clustered into a route, expected prediction error can be empirically estimated based on an independent validation data set not used during training, then applied to an independent test set to produce an expected prediction region with a user-defined level of expectation. Our results show that the prediction region contains the true interpolated future position at the expectation level set by the user, therefore producing a valid methodology for both estimating the future vessel location and for assessing anomalous vessel behavior.

THIS PAGE INTENTIONALLY LEFT BLANK

# TABLE OF CONTENTS

THIS PAGE INTENTIONALLY LEFT BLANK

# LIST OF FIGURES

# LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

# LIST OF ACRONYMS AND ABBREVIATIONS

| | |
|---|---|
| AI | Area of Interest |
| AIS | Automatic Identification System |
| COG | Course over Ground |
| CSV | Comma Separated Value |
| DBSCAN | Density-Based Spatial Clustering of Applications with Noise |
| ELM | Extreme Learning Machine |
| GP | Gaussian Process |
| IMO | International Maritime Organization |
| MDA | Maritime Domain Awareness |
| MLFF | Multi-layer Feed Forward |
| MMSI | Maritime Mobile Service Identity |
| NRO | National Reconnaissance Office |
| OU | Ornstein-Uhlenbeck |
| PAM | Partition Around Medoids |
| POI | Point of Interest |
| ROT | Rate of Turn |
| RSS | Residual Sum of Squares |
| SOG | Speed over Ground |
| UTC | Coordinated Universal Time |
| UTM | Universal Transverse Mercator |

THIS PAGE INTENTIONALLY LEFT BLANK

# EXECUTIVE SUMMARY

On any given day, roughly 6 million transmissions are communicated by 70 thousand ships that have an Automatic Information System (AIS) installed around the world. The transmissions amount to over two billion messages per year. The AIS system is a network of transceivers mounted on ships, and land-based stations, and satellites across the globe, and was originally intended to reduce collisions at sea. Since 2002, AIS transceivers are required to be installed on all ships that exceed three hundred tons, and on all passenger ships. The AIS data set captures key information about the ship consisting of the time-of transmission, latitude, longitude, speed, ship-type, and destination. The AIS data has been recorded and has since 2002 been used for many other reasons relating to maritime domain awareness.

Over the last ten years researchers have been seeking to predict where a vessel will be at some point in the future from the AIS data. This presents a challenge because the AIS data is messy. Much of the user input data such as destination or ship-type is either missing, incorrect, vague, or intentionally misleading. This fact makes the user input data fields appear to be unusable for the purpose prediction. The automated data fields such as time-of transmission, latitude, longitude, and speed, and course-over ground must be systematically cleansed to be useful in a predictive model.

Our sponsor, the National Reconnaissance Office (NRO), is looking for two main outcomes from this research. The first is to predict from the AIS data, the future location of a vessel. Second, the ability to identify anomalous behavior. And as a caveat, to do so in a way that requires minimal human intervention, and that can be applied anywhere. Using four months of the global AIS data from January through April 2014, we have produced a methodology, that brings them one step closer to attaining these goals.

A key idea in the current literature is that the series of vessel locations (tracks over time) can be represented by a network, where Points of Interest (POI) consist of ports, route intersections, and other static locations such as oil platforms. Given this network construct, analysts have strived to produced algorithms that accurately predict

future ship locations and associated prediction bounds along the routes connecting these POIs. These routes are non-linear and thus present a challenge for traditional data analysis techniques, especially considering that the quality of many of the data fields.

A major part of this thesis is the preparation and cleaning of the data to allow for effective implementation in a predictive model. We sort the data based on the unique identifier for the transceiver, then by time stamp so that the transmissions are in chronological order. The data is simply too big to be used at the global level, so we isolate a geographical area of interest (AI), then filter the data to the AI.

We clean the data fields that are useful for prediction. For example, we remove transmissions containing a reported speed of over three hundred miles per hour. The latitude and longitude fields were also occasionally infeasible and we systematically remove specific transmissions using an outlier detection algorithm. With cleaner data, we predict by looking at a specific route, and predict within that route, between POIs.

Although routes are not often defined by any boundaries like a road network, they can be extracted by grouping similar positional vectors produced by a moving vessel over time, using an algorithm called clustering. Once routes have been defined by the clustering algorithm, a route of interest can be extracted and future vessel positions can be predicted based on the route characteristics.

In the last step, we estimate a prediction region, or a latitude and longitude box, that the ship should be contained within at the 95 percent level of confidence. It is essential to convert from the latitude longitude coordinate system to the Universal Transverse Mercator (UTM) system which is measured in meters, and approximates the spherical earth by a series of interconnected flat surfaces. We divide outgoing sub-tracks into a training set, a validation set, and a test set based upon the vessel identification number. We then predict the validation set using two separate models, one for the future latitude value, and one for the longitude value. Once these predictions have been made we plot the numeric difference between the predicted locations and the true future locations (residuals) in meters. We extract the quantile (in meters) with the user defined

probability, then we apply these distances to the test set prediction to form a prediction region with the desired probability of containing the true future position.

We only use four automatically generated predictor variables in the models: speed, latitude, longitude, and course. The most important predictor variables turn out to be ones that we generated, that would be known at the time of prediction. The initial (naïve) prediction (Lat.hat, and Long.hat) is a constant-velocity linear model that projects the current position in a straight line along the current heading using $\text{distance} = \text{rate} \times \text{time}$. Used alone, this prediction is inaccurate on a curved route, but it is useful when included as another variable in our model.

We also use Reference Distances that are derived from 100 automatically generated points, distributed evenly through the geographical-space of the route. These are the great circle distances from each reference point to the current position of the vessel under consideration. When combined with organic variables, the naïve prediction along with the reference distances allow the random forest to choose which variables best predict the route based on the route and the time frame of the prediction. Our algorithm allows a user to enter the above information about a vessel at a point along a route, specify the time that they would like to predict into the future, and get a predicted location and the associated prediction region.

An interesting finding is that the random forest model allows us to predict along a curved route, while maintaining meaningful values of the prediction region bounds. Overall, our final model performs well across different regions and time periods and contained the true future vessel position with an accuracy rate of 94 percent, which is close to the targeted 95 percent containment.

THIS PAGE INTENTIONALLY LEFT BLANK

# ACKNOWLEDGMENTS

First, I would like to thank my wife, Tulin, and my children, William and Elizabeth, for allowing me the time to work on this thesis and coursework at NPS. I would like to thank Professor Robert Koyak for his invaluable input to this thesis, without which I might still be at square one. I would like to thank Sam Huddleston, Ph.D., for his input as well. Finally, I would like to thank the teaching staff of the NPS Operations Research Department for providing me with the tools to conduct this study and for giving me advice along the way.

THIS PAGE INTENTIONALLY LEFT BLANK

# I. INTRODUCTION

The International Maritime Organization (IMO) estimates that over 90 percent of the world's trade is carried by sea, as shipping continues to be the most cost-effective method to transport goods and raw materials globally (Tu, Zhang, Rachmawati, Rajabally, & Huang, 2016). Consequently, the safety and security of international sea lines of communication have perhaps never been more apparent. A growing demand for goods and materials around the world increases maritime traffic, which in turn increases the likelihood of collisions in congested areas, and presents more opportunities for piracy groups or terrorists to exploit. Harati-Mokhtari, Wall, Brooks, and Wang (2007) estimate that human error accounts for 80 to 85 percent of recorded maritime accidents. Irregular forces such as those who attacked the USS *Cole* in 2000 in Yemen are also of concern. An accurate point prediction of a vessel's future location can be useful to monitor traffic and to detect anomalies that could represent security threats. Because of uncertainties inherent in prediction, it is appropriate that predictions of location be accompanied by uncertainty regions that contain the true future location of a vessel within a certain level of tolerance.

As waterways have become increasingly congested, Maritime Domain Awareness (MDA) is becoming increasingly important to the U.S. Navy (Department of the Navy [DON], 2007). A key tool in maintaining MDA is the Automated Information System (AIS), a network of transceivers that provides information about the global movement of vessels at sea. Since 2002, the IMO has required that AIS transceivers be installed on ships over 300 tons, and on all passenger vessels, to increase safety of life at sea. Because AIS allows all vessel operators to see the location, heading, and speed of other ships in the surrounding area, collisions can be avoided, thus preventing both monetary loss and loss of life. Other benefits of AIS include traffic monitoring, search and rescue operations, accident investigations, navigational aid, and ship tracking (Balduzzi, Pasta, & Wilhoit, 2014). An example of how an AIS display may appear aboard a vessel is shown in Figure 1.

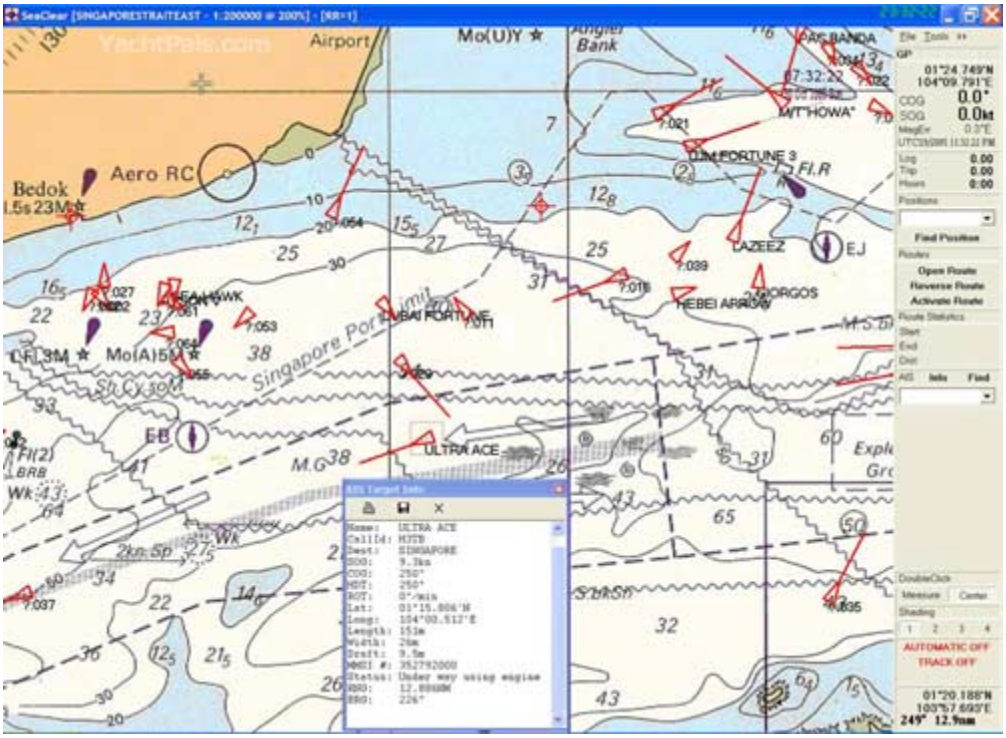



Figure 1. A Typical AIS Display. Source: Hampton (2009).

A vessel operator with AIS is able to get useful information about the other vessels in the area by selecting a vessel icon (depicted as triangles in Figure 1). Information such as speed, heading, latitude, and longitude aid the pilot in navigation. In addition to these basic features, other fields are updated by the pilot such as the destination, country of origin, and the current activity the ship is engaged in. An example of an activity undertaken and manually entered by the vessel operator might be “fishing” or “at anchor.” While these are useful features of AIS, the information is not always perfect. The user input data is often dubious and it may not be of great use for prediction of the future position of a vessel underway.

If an analyst possesses information on other vessels in the area that aid in a decision to avoid a collision, then how can he best represent this decision as an algorithm? Like a vessel operator, an algorithm must predict the future location of one or more vessels to prevent a collision. Similarly, when a vessel disappears, a search-and-rescue team must decide where to look which also involves predicting future vessel location. Vessel monitoring stations also would benefit from such an algorithm because

AIS transceivers only produce transmissions intermittently based upon the speed of a vessel.

Maritime security organizations also may benefit from an algorithm that predicts future vessel location and the uncertainty associated with that prediction to detect anomalous vessel behavior. If an analyst can automatically calculate an accurate point prediction for vessel location and a prediction region around that location, it might warrant investigation if a vessel is not contained in that prediction region. Pallotta, Vespe, and Bryan (2013) describe counter-piracy operations that depend on the ability to predict where commercial merchant traffic and pirate actions groups are likely to intersect. They note that merchant vessels often disable their AIS transceivers when transiting areas of high risk to piracy.

Anomaly detection is important for several reasons. First, it is useful for identifying potential security threats near populous coastal waterways. If anomalous behavior is identified early enough, it may be possible to react in time to prevent or limit damage. Additionally, the detection of an anomalous vessel might help to identify ships that have lost control or are having serious mechanical issues. If a vessel displays anomalous behavior, then it is doing something that is not defined by the established norms of the route. While anomalous behavior does not imply nefarious behavior, the ability to automatically detect anomalous behavior could aid security analysts in deciding how to best allocate limited resources to investigate potential threats.

In this thesis, we use two types of statistical learning models to predict the future locations of vessels at sea and to determine which method provides the best performance. Several methods have been developed over the last decade to address the problem of prediction in the maritime domain. Current methodologies for predicting vessel trajectories fall into three classes according to how they are implemented: physical-model based methods, learning-model based methods, and hybrid models (Tu, Zhang, Rachmawati, Rajabally, & Huang, 2016). Physical models consider all possible influencing factors and use physical laws of motion to predict the future trajectory of a vessel; however, this method is used primarily for building simulations. Learning models

use historical AIS data to develop a model of motion characteristics. A hybrid model may include both components of a physical model and historical motion data.

## A. RESEARCH OBJECTIVES

Because our purpose is to develop a prediction method that can be applied flexibly, with minimal need for tailoring to local conditions, we propose an approach that is non-parametric in nature and based solely on historical AIS data. First, we consider how to predict the future location of a vessel based upon AIS information and route characteristics of outgoing tracks from a port of origin. Second, we seek to estimate the bounds of a prediction region that is likely to contain the future location of the vessel. Third, we seek a methodology that is applicable across all regions. Finally, we seek a methodology that can be implemented automatically (requiring minimal human intervention). We construct our models using AIS data from around the world for the period of January 2014 to April 2014, and we investigate areas near the Port of Los Angeles; the Port of Barcelona, Spain; and the Port of Newark, New Jersey.

## B. THESIS STRUCTURE

The remainder of this thesis is organized as follows. In Chapter II we review current literature related to maritime navigation, prediction, and anomaly detection. We are particularly focused on the literature that uses AIS data. However, some research from related fields such as traffic management and aviation also is included. In Chapter III we describe the AIS data and the process of rearranging and transforming the data into a form that will allow us to use it in our algorithms. We will also describe the different modeling techniques used for point prediction as well as prediction-region construction. In Chapter IV we present the results of our models applied to each of the three regions analyzed. We present our conclusions and propose topics for additional research in Chapter V.

# II. LITERATURE REVIEW

This literature review describes how some of the methods that relate to this thesis have been implemented over the last decade. This review concludes with a discussion of how this thesis fills a gap in the current literature relating to prediction of future vessel location and anomaly detection using AIS data.

Khan, Cees, and Kaye (2005) use a multi-layer feed-forward (MLFF) neural network trained using singular value decomposition and genetic algorithms to predict the angle of ship (pitch) for up to 160 seconds. The authors cite inadequacies of other methods such as autoregressive moving average models and Kalman filters to calculate accurate short-term estimates of a ship's state in rough seas to enable safer landing of aircraft on a ship.

Palacios and Doshi (2008) use a neural network to predict future position of an aircraft using two different approaches. In their X/Y approach the same type of neural network is applied twice, once to predict the future longitudinal coordinate, and once to predict the latitude coordinate. To implement this method, they choose to use distance traveled in the last few seconds to predict thirty seconds into the future. Their second method, called the bearing/distance approach, is based on estimating the direction of movement and the distance the aircraft will travel. Then they calculate future position using trigonometry. The authors found the bearing approach to be 5 percent to 10 percent less accurate than the X/Y approach.

Morris and Trivedi (2008) represent the learning of paths between different Points of Interest (POI) in video surveillance as a three-step process. Tracks created by moving objects are first preprocessed using a form of dimensionality reduction such as linear interpolation to put the tracks on a comparable basis. Second, the tracks are clustered to represent different routes that are comprised of similar trajectories. Finally, the routes are modeled using either the whole route or by breaking the route into segments. In the final step, they summarize their centroid method for minimally specifying a route as well as an extension to the centroid method called an envelope which specifies the variation along a

route. Methods to implement the envelope method currently in use are the extreme point method and the Gaussian distribution method.

Ristic, La Scala, Morelande, and Gordon (2008) use historic AIS data to extract motion patterns which are then used to construct motion anomaly detectors using adaptive kernel density estimation. They then use the anomaly detector sequentially on incoming AIS data to detect anomalies under the null hypothesis that there is no anomaly. Additionally, the authors use historic motion pattern data to predict the motion of vessels using the Gaussian sum tracking filter.

Zhu (2011) discretizes a region of interest using hash codes and uses association rules to extract knowledge of highly traveled grids. The author also uses the association rules to say with a certain confidence that if grid “x” and grid “z” have been visited, then grid “y” will also have been visited.

Morris and Trivedi (2011) use Gaussian mixture modeling to find points of interest, and use trajectory clustering to form routes and use hidden Markov models to predict future location of vehicles moving at intersections and detect anomalous trajectories.

Vespe, Visentini, Bryan, and Braca (2012) model vessel behavior as a network of waypoints (entry and exit points, turn points, or stop points) and sea lanes. They define a route as a sequence of sea lanes that are each characterized by statistical properties including Course Over Ground (COG), Speed Over Ground (SOG) and spatial deviation from the segment. Finally, the authors demonstrate the waypoint identification model effectiveness in an area of high terrestrial AIS coverage (Adriatic Sea) and low terrestrial AIS coverage (Red Sea and Gulf of Aden).

Pallotta, Vespe, and Bryan (2013) mention that the use of turn-points in the “vectorial” or network model does not work well for unregulated traffic areas. They propose a density-based algorithm (DBSCAN) to derive stationary areas and entry and exit points and derive route objects between these points using the vessel flow vectors which incorporate vessel turn. Routes are created by clustering the vessel flows from one POI to another. Additionally, they propose a method of predicting future location of a

vessel based on a sequence of circles of a user defined radius centered on the observed positions. The authors mention that a drawback to using the circle method is that the chosen radius $d$ could be too small for the route resulting in the characterization of the local route behavior to be based on a reduced number of neighbors. If the radius $d$ is too large, then the characterization would be biased by non-rectilinear routes. They claim that a radius "on the order of a few nautical miles" is effective for any route.

McAbee (2013) uses the Hough transformation to extract normal linear traffic patterns from AIS data to generate normal sea-lanes in both open-ocean and coastal areas. Once the sea-lanes have been defined, the lane is broken into sections along the direction of travel. A normal distribution is fit to each section of the lane to account for heteroscedasticity and those vessels observed outside a user defined threshold are determined to be anomalous.

Tester (2013) uses k-means clustering to group vessels with similar course and speed to classify ship movement then tracked cluster membership by comparing the distance between vessels over time; however he does not predict future ship location.

Stone, Streit, Corwin, and Bell (2014) illustrate the tracking of a surface ship using a Kalman filter. They specify a motion model using an Integrated Ornstein-Uhlenbeck (IOU) process. The authors then specify a measurement model assuming that the relationship between the measurement and the target state is linear. They then simulate measurements using their measurement model and apply a continuous-discrete Kalman filter recursion to obtain tracker output. The authors conclude that these assumptions are optimal for a Kalman filter, but a motion model based on the Ornstein-Uhlenbeck process "is not a good representation of the actual motion of ships" (p. 10).

Pallota, Horn, Braca, and Bryan (2014) present a method to predict future vessel location based upon the Ornstein-Uhlenbeck (OU) stochastic process. The model parameters are estimated from recurrent route patterns contained in the AIS data, where routes are the arcs between points of interest. First, the authors assume that a vessel has been classified correctly to be a member of a specific route. Next, they assume that the vessel dynamics are represented by a set of linear stochastic differential equations. Three

different parameters are estimated that characterize the statistical properties of the route. The key benefit of the OU method is that the variance of the vessel position grows linearly with time as opposed to a higher nonlinear rate in other previous models. Data are converted from latitude and longitude to Universal Transverse Mercator (UTM) coordinates. The authors present results for three cases for which the prediction error standard deviation is on the order of 1000 meters at a prediction time interval of five hours.

Millifiori, Braca, Bryan, and Willett (2016) continue the previous work of Pallotta et al. (2013) with a focus on vessels that travel without maneuvering as might occur in open sea. The authors find that the nearly constant velocity (NCV) model may be unrealistic for most vessel traffic scenarios since vessel operators vary the speed frequently. In addition, they present evidence to suggest that non-maneuvering vessel velocity follows an OU process, and consequently that vessel position is represented by an integrated OU process (IOU). Their results show that the standard deviation of the prediction error along the x and y coordinates are on the order of 3 km after five hours.

Mao et al. (2016) use an extreme-learning machine (ELM) to predict future vessel location based upon AIS data off the coast of California. After selecting a route, they use the latitude, longitude, SOG, COG, Rate of Turn (ROT), time, and Maritime Mobile Service Identity (MMSI) as the data to use in their experiment. The authors calculate an error distribution of between 0 and 2.5 nautical miles for a 20-minute prediction, and between 0 to 6 nautical miles for a 40-minute prediction.

Tu et al. (2016) describe three of the most commonly used modeling methods in use including physical models, learning models, and hybrid models. The authors note that physical models may be practical for implementation aboard an individual ship or in a simulation, although the detailed information required to fit these models is not likely to be available for other vessels, as is the case with the AIS data set. They note that the neural-network approach is particularly good at fitting complex functions, but the training process can be slow to convergence and there are no general rules about how to choose the number of hidden layers, number of neurons, or activation function. The Gaussian Process (GP) method is also described as a powerful tool for predicting ship trajectories.

The authors also mention the use of OU processes in which the assumption of stationarity is assumed (no change in mean trajectory vector or variance) and they note that this is a restrictive assumption for real-world applications. The authors mention that there is potential benefit to combining physical models with learning models to achieve better prediction outcomes.

Bay (2017) uses AIS data in the area of Port Fourchon, LA to examine the effectiveness of clustering for identifying navigation routes in the northern Gulf of Mexico, and to measure the effects of weather and sea-state on navigation. The Gulf of Mexico near Port Fourchon has many oil and natural gas platforms that are serviced by vessels that are based at Port Fourchon, which makes it difficult to segregate tracks into a small number of clusters. Her research is aimed at identifying factors that could be useful in building better prediction models.

The studies reviewed here use clustering of similar trajectories to detect sea routes and use dispersion of vessel positions within these routes to develop prediction regions for future movement of vessels. Some authors use neural network models in a time-series context for this purpose. We aim to group similar vessel trajectories to define routes, then we use the collective information about a route from all ships that have traversed that route to predict the future position of a ship at some time in the future. This method considers where each vessel is located at any point in a route and then predicts where those vessels will be at $t$ minutes into the future taking all the relevant information into consideration. Additionally, we propose a methodology for constructing prediction regions using the AIS data. We do not make any distributional assumptions about the route.

THIS PAGE INTENTIONALLY LEFT BLANK

# III. DATA COLLECTION AND PREPARATION

## A. DATA DESCRIPTION

The AIS data used in this thesis covers the entire world geographically, during the period January through April 2014. The data consist of records of two types: static and dynamic. The fields in the dynamic and static records are described in detail by the U.S. Coast Guard Navigation Center (2017) and are presented in the Appendix.

Dynamic records are automatically transmitted using a vessel's AIS transceiver, and consist of the motion-related information that changes as a vessel moves in space and time. A new dynamic record is transmitted every 2 to 10 seconds while a vessel is underway and every three minutes while at anchor. For a typical day, the AIS data contains approximately six million dynamic records. The Maritime Mobile Service Identity (MMSI) is a unique number used to identify a specific AIS transceiver, and usually stays with a vessel. The latitude and longitude fields together represent the location of a vessel at the time of transmission. Course over ground is the angle relative to true north that the vessel is traveling at the time of transmission from 0 to 359 degrees. Heading is the magnetic compass angle that the vessel is traveling from 0 to 359 degrees. The navigational status field denotes whether the vessel is underway, at anchor, or fishing for example. Speed over Ground (SOG) represents the vessels speed in knots. Each transmission also contains the time stamp containing the date, hours, minutes, and seconds in Coordinated Universal Time (UTC). We find that the dynamic record fields Latitude, Longitude, COG, and Speed are the most useful in prediction. In particular, the Speed field must be checked for obvious measurement error, and those records must be removed. The Latitude and Longitude coordinates must be checked as well to ensure that the reported distance travelled over time does not exceed what is physically possible for a ship.

The static records describe non-motion related attributes of a vessel and are updated every six minutes. Table 4 in the Appendix lists all the fields that are transmitted in a static record. Because static records are not automatically transmitted but require

human intervention, the static reports are subject to human error and are not regarded as highly reliable. The IMO number represents a specific vessel by a unique identifier that never changes. An installer manually enters the IMO number at the time of installation. We do not use the IMO number in this thesis, although it may be useful for intelligence analysis. The destination field is intended to allow the vessel operator to communicate his next destination. The destination field might appear at first to be useful in predicting where a vessel might be heading, however it is often incorrect and sometimes intentionally misleading. The vessel-type field that classifies a vessel as a cargo ship, passenger vessel, tanker, etc. also is not reliably recorded and often is ambiguous. The dimensions of the ship may also be calculated from fields that report length from bow to stern and length from port to starboard in meters, but these too are often unreliable.

Many of the fields provided in an AIS report are not valuable for predicting future location either because the data is manually input (static data), or because the transceiver itself does not report automated data accurately. Harati-Mokhtari et al. (2007) discuss the challenges of implementing AIS globally and of relying upon many manufacturers to produce standardized AIS transceivers. The authors cite two studies in which errors were summarized according to the data field. They find that errors in AIS data are not uncommon:

- approximately two percent of MMSI field entries are erroneous;
- approximately 74 percent of the vessel type field entries are vague or misleading;
- approximately 30 percent of the navigational status entries are incorrect;
- approximately 47 percent of the vessel length field entries are incorrect;
- approximately 18 percent of the beam field entries are incorrect;
- approximately 49 percent of the destination field entries are erroneous or even intentionally misleading.

Although the authors do not analyze the position fields in detail, they mention a study that finds one percent of the sampled data showing latitude of more than 90°, longitude of more than 180°, or the position 0°N 0°W, which obviously are incorrect.

## B. DATA PROCESSING

Bay (2017) provides a description of the process by which the AIS records are parsed into an analyzable form; we use the same process in this thesis. AIS records are originally transmitted in the AIVDM/AIVDO format and converted using a regular expression based script. The output of this process is a Comma Separated Value (CSV) file. The CSV files are then converted into a spatial-points data frame in R (R Core Team, 2016) using the "sp" package (Pebesma & Bivand, 2005). Our data covers the period of January 1, 2014 to April 30, 2014 hereafter referred to as "the period of interest" unless otherwise specified.

We convert speed from nautical miles per hour (knots) to meters per minute for our models because we use minutes to designate the time period for which we are predicting into the future. Additionally, we approximate the size of the ship by using the distance to bow and stern and the distance from port to starboard of the transceiver in the following manner:

$$ShipSize = (dbow + dstern) \cdot (dport + dstar) \tag{1.1}$$

Finally, we note that it is necessary to convert coordinates from latitude and longitude to Universal Transverse Mercator (UTM) system of eastings and northings which are measured in meters. The UTM system projects the roughly spherical earth into a series of flat surfaces that approximate the surface of the earth and may be thought of as a disco-ball representation of the earth. This procedure allows us to estimate the residual error of our predictions in meters (as opposed to degrees).

We find that is critical to clean the data of anomalous reports prior to running our models. Some of the reported speeds are simply infeasible (e.g., more than 2,000 meters per minute) and we remove these observations. The reported coordinates can sometimes be misleading; in some cases the distance traversed by the ship in a given period of time exceeds what is feasible and we remove these observations as well. For example, if a ship's coordinates place it 10 km from the last reported position in thirty seconds, and the next position is back within a feasible range, then we treat the extreme report as an error and remove it. Koyak (2017) explains the algorithm used to perform outlier identification:

Our approach to outlier detection is to begin by evaluating the expression "observation $r$ is anomalous with respect to observation $s$" with respect to every pair of measurements in a track. We address anomaly criteria below; assume for now that a criterion has been adopted and that the anomaly relationship is symmetric. More precisely, let $a(r,s)=1$ if $r$ and $s$ are anomalous and $a(r,s)=0$ otherwise; symmetry implies that $a(r,s)=a(s,r)$. If $a(r,s)=1$ either one or both of observations are potential outliers, but which of the two should be treated as such cannot be resolved using this information alone.

Let A denote the matrix of anomaly indicators $a(r,s)$ and let $b$ denote the vector of its row sums. Suppose that observation $r$ is an outlier and that is the only one present in the track. Because we expect it to be anomalous with respect to many if not all of the other observations $b(r)$ should be large, while $b(s)=1$ for all $s \neq r$. Similarly, if there are multiple outliers the values of $b(r)$ should be large for those observations and small for the non-outliers. (p. 8)

The pseudo-code of the track-outliers algorithm is depicted in Figure 2.

```
Input :      A (n × n matrix of anomaly indicators)
Define :     b (row sums of A, an n – vector)
             o (outlier indictors, an n – vector)
Initialize : o(j) = 0, j = 1 : n
while (max(b(j), j = 1,…,n) > 0) {
    r ← argmax(b(j))
    o(r) ← 1
    b(r) ← 0
    for (j = 1,…,n){
        if (o(j) == 0)
            b(j) ← b(j) – a(r, j)
    }
}
return(o)
```

Figure 2. Pseudo-code for the Track-Outliers Function. Source: Koyak (2017).

## C. CONVERTING THE RAW DATA TO USABLE FORM

Next, we arrange the AIS data into an appropriate form to be used in a predictive model (such as regression). That is, our AIS data contains a response column and several possible predictor variables. We merge the dynamic data with the desired fields from the static data (by matching the MMSI field) into one spatial points data frame that covers the period of interest. For example, we include the ship type and the calculated ship size to the data frame to determine if it is a valuable predictor in our model. We then merge the data frame and filter based upon a geographical area of interest (bounding box) defined by a set minimum and maximum latitude and longitude values. This rectangle contains the trajectories of ships in the area distinguished by their MMSI. We select a port of interest and find the coordinates for the port, then filter the data based upon whether a ship has traversed within a specified range of the port. If a vessel is ever within the specified range of the port then it is considered to have arrived at or departed from the port. We define stopping criteria that are based upon distance travelled and time. We use the haversine distance which is based on a spherical model of the earth defined by:

$$d((x_1, y_1), (x_2, y_2)) = 2r_0\sqrt{\sin^2\left(\frac{y_2 - y_1}{2}\right) + \cos(y_1)\cos(y_2)\sin^2\left(\frac{x_2 - x_1}{2}\right)}. \qquad (1.2)$$

In this formula $x_i$ refers to the first and second longitude values, $y_i$ refers to the latitude values, and $r_0$ refers to the approximate radius of the earth (6,378,137 meters). The haversine distance can be calculated in R using the "distHaversine" function in the package "geosphere" (Hijmans, 2015). We use the haversine distance because the distance between two points on the surface of the earth is not a straight line, but an arc that tracks the earth's curvature.

Over the course of time, a vessel makes a variable number of trips between pairs of stop points. We are interested in extracting instances of travel that we call sub-tracks that isolate movements relative to a specific POI and a single origin or destination. Sub-tracks are identified by a vessel being stopped at the POI and also stopped at a different point after having left the POI (outgoing sub-track) or before having arrived at the POI (incoming sub-track). We identify that a vessel is stopped if it exhibits little motion over

a substantial period of time. If (distance travelled)/$\Delta t$ is less than a speed threshold , where $\Delta t$ is an interval of time specified (e.g., 20 minutes) and the speed threshold is a speed below which the vessel is assumed to be stopped (e.g., 25 meters per minute), then the vessel is considered to have arrived at a stop point and the sub-track is terminated. A sub-track is classified as incoming or outgoing depending upon whether the distance from the POI is increasing or decreasing.

We only consider outgoing sub-tracks relative to a POI because any sub-track can be considered as an outgoing sub-track from some POI. We use linear interpolation at user-specified "odometer" distances travelled to standardize the sub-tracks to be used in a clustering algorithm. The "odometer" distance of a vessel is calculated as the sum of the distances travelled by the ship since it left the port as reflected in the AIS reports, and not the distance between the port and the current location of the vessel. Figure 3 shows a representation of standardized sub-tracks.

Filter

| | Subtrack | ShipType | ShipSize | Long1 | Lat1 | Long2 | Lat2 | Long3 | Lat3 | Long4 | Lat4 | Long5 | Lat5 | Long6 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 4 | Tug | NA | -74.06 | 40.63 | -74.04 | 40.59 | -74.02 | 40.54 | -73.97 | 40.51 | -73.92 | 40.49 | -73.87 |
| 3 | 6 | NA | NA | -74.06 | 40.63 | -74.04 | 40.59 | -74.02 | 40.54 | -73.97 | 40.51 | -73.92 | 40.49 | -73.86 |
| 4 | 9 | Cargo ship | 6688 | -74.06 | 40.64 | -74.04 | 40.59 | -74.02 | 40.54 | -73.98 | 40.51 | -73.92 | 40.49 | -73.86 |
| 5 | 11 | NA | NA | -74.06 | 40.64 | -74.04 | 40.59 | -74.02 | 40.54 | -73.97 | 40.51 | -73.92 | 40.49 | -73.86 |
| 6 | 13 | Cargo ship | 8320 | -74.06 | 40.64 | -74.04 | 40.59 | -74.02 | 40.54 | -73.98 | 40.51 | -73.92 | 40.49 | -73.86 |
| 7 | 15 | NA | NA | -74.06 | 40.63 | -74.04 | 40.59 | -74.02 | 40.54 | -73.98 | 40.51 | -73.92 | 40.49 | -73.86 |
| 8 | 17 | NA | NA | -74.06 | 40.64 | -74.04 | 40.59 | -74.02 | 40.54 | -73.97 | 40.51 | -73.92 | 40.49 | -73.86 |
| 9 | 19 | Cargo ship | 9408 | -74.06 | 40.63 | -74.04 | 40.59 | -74.02 | 40.54 | -73.98 | 40.51 | -73.92 | 40.49 | -73.87 |
| 10 | 21 | NA | NA | -74.06 | 40.64 | -74.04 | 40.59 | -74.02 | 40.54 | -73.97 | 40.51 | -73.92 | 40.49 | -73.86 |

A sample of standardized sub-tracks (using linear interpolation).

Figure 3. Standardized Sub-tracks Near the Port of Newark, NJ

The positions in Figure 3 are interpolated every 5 kilometers (odometer distance) up to a distance of 400 kilometers.

Our proposed prediction method uses a number of techniques from modern data analysis including cluster analysis, neural networks, and random forests. We briefly

describe these techniques below, and show how they may be applied to the AIS data to obtain predictions of future vessel location and the associated prediction regions.

## D. CLUSTER ANALYSIS

Cluster analysis (clustering) has been used by scientists for decades to systematically find groups in their data. The objective of clustering is to place objects in groups called clusters that share similar characteristics, and to produce clusters that are as dissimilar from one another as possible. In our case a route contains $n$ trajectories (sub-tracks) consisting of $p$ latitude-longitude coordinate pairs and is placed in an $n \times 2p$ matrix represented as follows:

$$\begin{array}{c} \\ n \text{ objects} \end{array} \begin{array}{c} \overset{2p \text{ variables}}{\overbrace{\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad}} \\ \begin{bmatrix} (x_{11}, y_{11}) & \cdots & (x_{1f}, y_{1f}) & \cdots & (x_{1p}, y_{1p}) \\ \vdots & & \vdots & & \vdots \\ (x_{i1}, y_{i1}) & \cdots & (x_{if}, y_{if}) & \cdots & (x_{ip}, y_{ip}) \\ \vdots & & \vdots & & \vdots \\ (x_{n1}, y_{n1}) & \cdots & (x_{nf}, y_{nf}) & \cdots & (x_{np}, y_{np}) \end{bmatrix} \end{array}. \tag{1.3}$$

The next step in clustering is to calculate the distance between objects to quantify the dissimilarity between each object. Although there are several choices, we use the sum of haversine distances at the interpolation points:

$$d(i, j) = \sum_{f=1}^{p} \text{distHaversine}((x_{if}, y_{if}), (x_{jf}, y_{jf})). \tag{1.4}$$

These inter-object distances are then placed in an $n \times n$ distance-matrix **D**.

Now that the distance (or dissimilarity) matrix has been calculated, there are two broad classes of clustering algorithms that can be used: partitioning, and hierarchical. The partitioning method divides the observations in to $k$ clusters, where $k$ is chosen by the user. Each cluster must contain at least one object, and each object must belong to exactly one group. We use Partitioning Around Medoids (PAM) because it generalizes k-means clustering by not assuming that the clusters represent normal distributions with a common covariance matrix (Kaufman & Rousseeuw, 1990). The PAM algorithm begins by arbitrarily designating $k$ representative objects (medoids) in the data set, after which the

remaining objects are assigned to the nearest medoid. Medoids are selected iteratively such that the average distance between the medoid and all other objects in a cluster is minimized (Kaufman & Rousseeuw, 1990).

Kaufman and Rousseeuw (1990) measure the strength of clustering using a metric called the silhouette coefficient which is calculated as follows:

- For an object $i$ from the data set where $A$ is the cluster to which it has been assigned:

  $a(i) =$ average dissimilarity of $i$ to all other objects of $A$. (1.5)

- For any other cluster $C$ that is different from $A$:

  $d(i,C) =$ average dissimilarity of $i$ to all objects of $C$ . (1.6)

- After computing $d(i,C)$, for all clusters $C \neq A$ , pick the smallest of those:

$$b(i) = \min_{C \neq A} d(i,C) \tag{1.7}$$

- If $B$ is the cluster which is second best to cluster $A$, the silhouette coefficient for object $i$ is calculated by:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \tag{1.8}$$

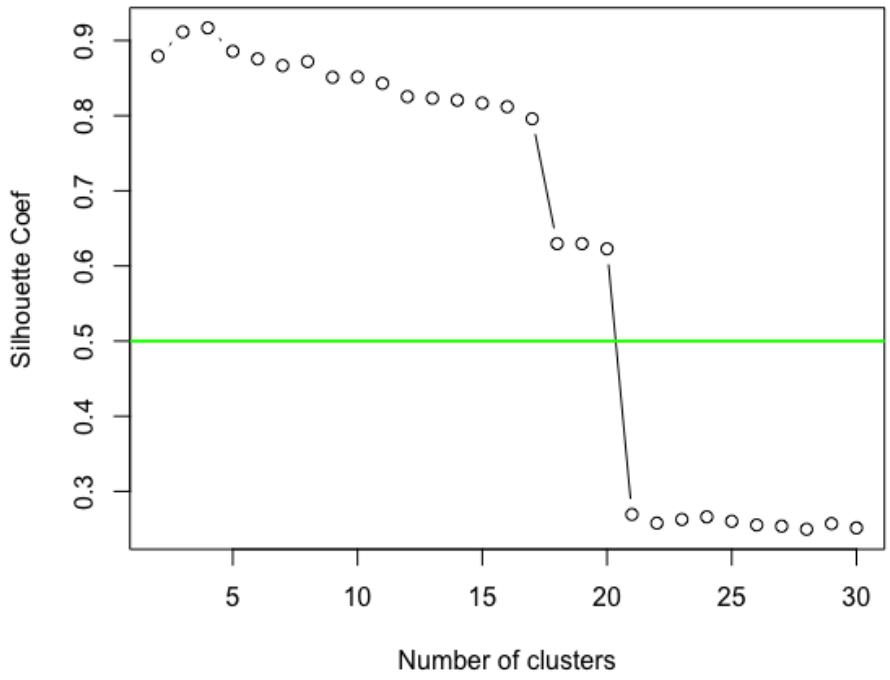
- The silhouette coefficient of a cluster is the average of the silhouette coefficients of all objects within that cluster. The silhouette coefficient for the cluster solution with $k$ clusters, $S_k$, is the average of all silhouette coefficients over the data set for that solution.

Kaufmann and Rousseeuw (1990) suggest picking the number of clusters $k$ to maximize $S_k$. If the value of $S_k$ is between .71 and 1 then a strong structure has been found, if $S_k$ is .51 and .70 then an acceptable structure has been found, and if $S_k$ is less than or equal to .50 then the structure is weak or non-existent.

Now we address how the PAM clustering algorithm applied to the AIS data to extract routes (groups of similar trajectories). Our goal is to find routes that start and end at different POIs so that one may use the information from the AIS data points to

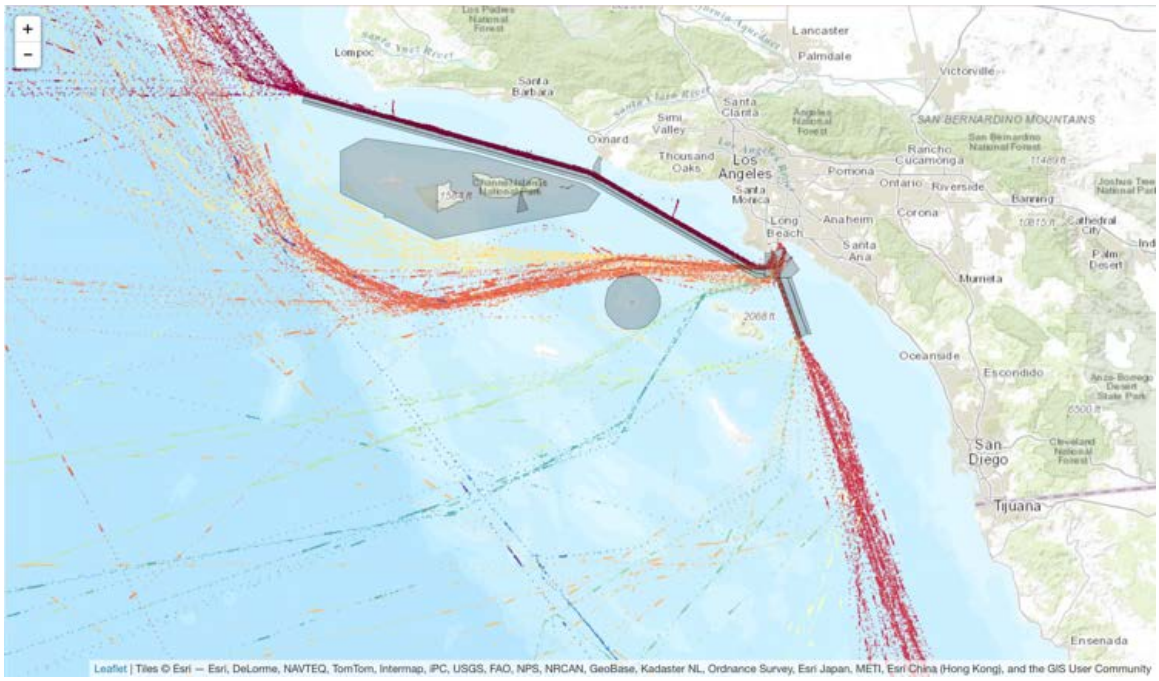
characterize the route and make better predictions within that clustered route. An analyst may use this route data to characterize the uncertainty of his predictions.

We use the PAM algorithm from the "cluster" package in R (Maechler, Rousseeuw, Struyf, Hubert, & Hornik, 2016) to cluster standardized routes using medoids based only upon the interpolated latitudes and longitudes obtained in the previous step. We run PAM to calculate the average width of the silhouette coefficient for different values of $k$. We then use Kaufman and Rousseuw's (1990) guidelines for choosing $k$, that is to use values of $k$ such that the silhouette coefficient is greater than .5 and look for the "knee in the curve" if present. As an additional cluster quality control, we use members of the desired cluster that have a silhouette value greater than .5 to filter weak members of the cluster. We also assess the quality of the cluster visually by plotting the clustered sub-tracks and ensuring that the clusters are of similar sub-tracks. Figure 4 shows the results of cluster analysis in the Newark, NJ area and Figure 5 shows a visualization of the results of clustering plotted on a map.




These clustering results suggest that $k$ = 20 clusters is appropriate

Figure 4. Results of Clustering Sub-tracks Near Newark, NJ




Results of clustering the standardized sub-tracks (clusters are color coded). This plot and all other map plots were rendered using the "leaflet" package in R (Cheng, J., Karambelkar, B., & Xie, Y., 2017).

Figure 5. Plot of Outgoing Sub-track Clusters Departing Los Angeles

For this thesis, we choose a cluster (route) with the highest number of sub-track members to perform analysis because most routes do not have sufficient observations to support statistical analysis over the time frame considered. In the case of the Figure 5, we choose the dark-red cluster that passes near Oxnard because it contains the largest number of sub-tracks which ensures that it can be used effectively in an analysis. Interestingly, the ships closely follow the overlaid shipping lanes depicted in black and gray. After clustering has been performed, the regional AIS data is filtered based upon the cluster membership.

## E. INTRODUCTION TO NEURAL NETWORK

A neural network prediction model is inspired by the manner in which the human brain learns. The components of a neural network consist of a network of neurons connected by synapses that can be represented in three (or more) layers. The input layer consists of the attributes of the object of interest and connects to one or more hidden layers. The hidden layer(s) then connect to the output layer consisting of attributes that

one seeks to classify or estimate: the dependent variable. Hastie, Tibshirani & Friedman (2009) provide an overview of neural networks and we will briefly summarize here using their notation. A visualization of a simple form of neural network in Figure 6.

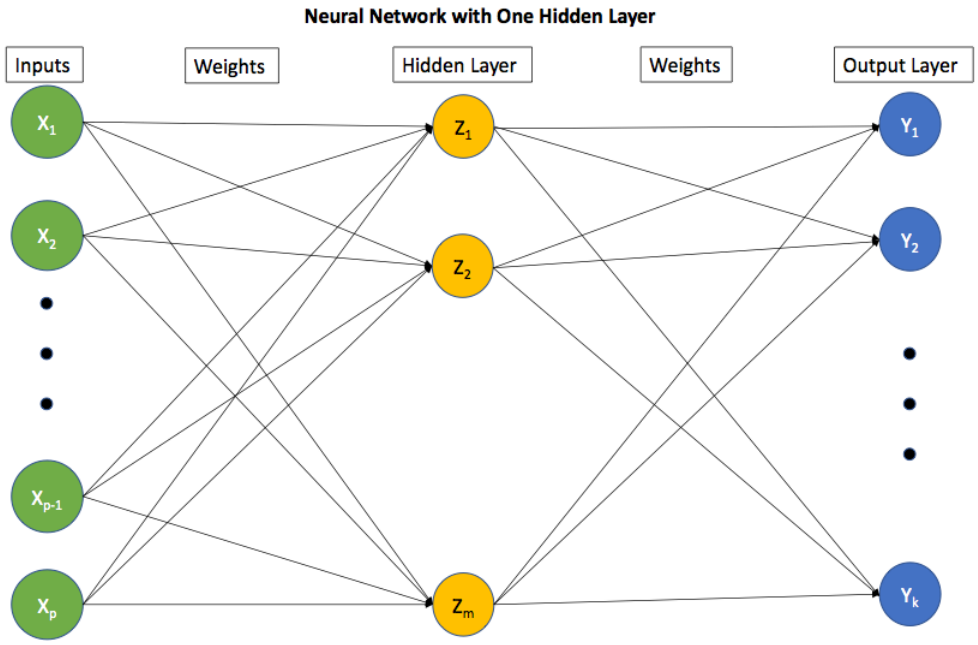



Figure 6. Depiction of a Simple Neural Network

The orange circles in Figure 6 represent derived features (neurons) that take a set of inputs $X = \{x_1, x_2, \cdots, x_p\}$. The set of derived features (neurons) $Z = (Z_1, Z_2, \ldots, Z_M)$ of the hidden layer are calculated as linear combinations of the inputs, and the target $Y_k = f_k(X)$ is modeled as a function of linear combinations of the $Z_m$ as follows:

$$Z_m = \sigma(\alpha_{0m} + \alpha_m^T X),\ m = 1, \ldots, M\ , \tag{1.9}$$

$$T_k = \beta_{0k} + \beta_k^T Z,\ k = 1, \ldots, k\ , \tag{1.10}$$

$$f_k(X) = g_k(T),\ k = 1, \ldots, K\ . \tag{1.11}$$

The function $\sigma(v)$ in equation 1.9 is called the activation function. Some commonly used activation functions are the sigmoid function, the hyperbolic tangent function, and the rectified linear activation function. The sigmoid activation function is formulated as:

$$\sigma(v) = \frac{1}{1+\exp(-v)} \ . \tag{1.12}$$

The complete set of weights (the unknown parameters) which the authors denote as $\theta$ are:

$$\begin{matrix} \{\alpha_{om}, \alpha_m;\ m = 1, 2, \ldots, M\}:\ M(p+1) \text{ weights, and} \\ \{\beta_{0k}, \beta_k;\ k = 1, 2 \ldots, K\}: K(M+1) \text{ weights.} \end{matrix} \ , \tag{1.13}$$

The terms $\alpha_{0m}$ and $\beta_{0k}$ represent the intercepts of the model and are set to a constant value of 1. When a neural network is used for regression as in this thesis, $g_k(T) = T_k$.

To form a neural network, several neurons can be connected so that the output of one neuron can be the input of another neuron. When performing regression, the measure of fit is calculated using a loss function such as the sum-of squared errors:

$$R(\theta) = \sum_{k=1}^{K} \sum_{i=1}^{N} \left( y_{ik} - f_k\left(x_i\right) \right)^2 \ . \tag{1.14}$$

The authors describe how learning takes place by finding the optimal weighting scheme, one that minimizes $R(\theta)$ through a form of gradient descent optimizations called back-propagation when used in a neural network. For the squared-error loss function in equation 1.14 back-propagation is accomplished using a two-pass algorithm. The first pass is called the forward pass, where the current weights are fixed and the predicted values $\hat{f}_k(x_i)$ are calculated using equation 1.11. In the backward pass, the errors are calculated for the output nodes and the hidden layer nodes. Both sets of errors are used to calculate the gradients with respect to each weight, and then the weights are updated in a manner that reduces the loss function.

## F. INTRODUCTION TO RANDOM FORESTS

To illustrate how a random forest functions, we must first introduce the concept of a partition tree. A partition tree can be used to perform regression or classification, but we use it for regression in this thesis. Hastie et al. (2009) explain how tree based methods and random forests partition the feature space and we will summarize their work here using their notation. Tree based-methods "partition the feature space into a set of rectangles, and then fit a simple model (like a constant) in each one" (Hastie et al., 2009). Figures 7 and 8 illustrate how a feature space with two predictor variables is partitioned.

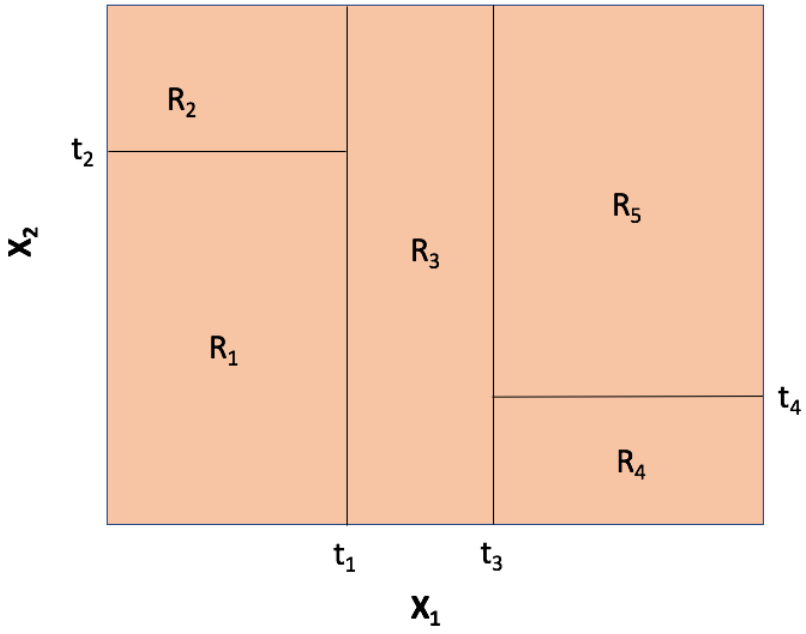



Figure 7. Example of a Partitioned Feature Space

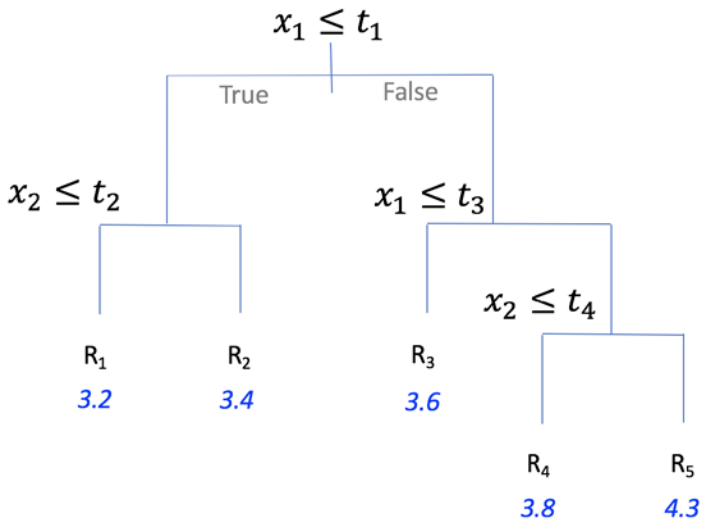



Figure 8. Partition Tree Example

Figure 8 shows a tree with four internal nodes denoted by $x_j < t_k$. The left branch of the internal node corresponds to when the statement is true, and the right branch corresponds to $x_j \geq t_k$. The regions and numbers at the bottom of the tree are called terminal nodes leaves, and are calculated by taking the mean of the response variable for the observations that fall in each region. This process divides the predictor space into five regions corresponding to the leaves on the tree as depicted in the Figure 7:

Suppose, as Hastie et al. (2009) illustrate, that our data contains $N$ observations and $p$ input variables and we wish to partition the feature space into $M$ distinct and non-overlapping regions. Let $x_i = \{x_{i1}, x_{i2}, \cdots x_{ip}\}$ be the set of $p$ inputs for each observation $i$, let $Y = \{y_1, y_2, \ldots, y_N\}$ be the set of all responses, let $R = \{r_1, r_2, \cdots r_M\}$ be the set of $R$ distinct and non-overlapping regions, and let $c_m$ be a modeled response constant, the average of $y_i$, in each region. Then the predictor takes the form:

$$f(x) = \sum_{m=1}^{M} c_m I(x \in R_m), \tag{1.15}$$

where $I(A)$ is equal to 1 if condition $A$ is true, and 0 otherwise. The goal is to minimize the residual sum of squared errors (RSS):

$$RSS = \sum_{i=1}^{N} (y_i - f(x_i))^2, \tag{1.16}$$

where $f(x_i)$ is the mean of the training observations that fall within region $R_m$. To determine the best binary partition, start with all the data and choose a splitting variable $j$, a split point $s$, and split the feature space into two half planes. Each region can be represented by:

$$\begin{aligned} R_1(j,s) &= \{X \mid X_j \le s\}, \text{ and} \\ R_2(j,s) &= \{X \mid X_j > s\}. \end{aligned} \tag{1.17}$$

The feature space is best partitioned by selecting the splitting variable $j$ and a split-point $s$ minimizes the sum of the RSS in each of the two separate regions. Now the two regions are again split using the same process into as more partitions. For a detailed description of how the best individual tree size may obtained, see (Hastie et al., 2009).

Decision trees by themselves are known to have drawbacks such as overfitting and sensitivity to outliers which leads us to random forests. Individual trees are characterized by high variance, but have a low bias. Random forests also partition the feature space, but do so in a manner that is less variable than an individual tree. A random forest works by creating many uncorrelated trees created by bootstrapping different versions of the training data. These trees are uncorrelated because at each split $s$ randomly selected predictor variables are used from the full set of $p$ predictor variables to make the split. Typically, $m \approx \sqrt{p}$ is a suitable number of predictor variables to use at each split. Finally, the prediction results from all the trees are averaged to produce a model with the best performing splits based on the RSS.

## G. IMPLEMENTATION OF A NEURAL NETWORK PREDICTOR

Our objective is to predict the position of vessel $\Delta t$ minutes into the future from a given point in time. We use data on sub-tracks in a cluster off the coast of Los Angeles during the period of January through April 2014. We again emphasize that we have transformed the AIS data and arranged it in a manner such that we can estimate a neural network from the data. We use a spatial-points data frame that has been put into UTM

format based on the zone for the region of interest. An example of clustered route data for Los Angeles is depicted in Figure 9.

| | Speed | Lat | Long | Course | AvSpeed | RefDist1 | RefDist2 | | RefDist100 | Lat.hat | Long.hat | FutLat | FutLong |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 373.4871 | 3732133 | 359800.7 | 300.7 | 454.6902 | 183952.3 | 163380.1 | | 76307.36 | 3735947 | 353377.8 | 3735820 | 353291.8 |
| 2 | 373.4871 | 3733977 | 356531.9 | 300.5 | 454.6902 | 180663.5 | 160096 | | 74631.23 | 3737768 | 350095.8 | 3737743 | 350056.1 |
| 3 | 373.4871 | 3735887 | 353172.8 | 300.2 | 454.6902 | 177303.1 | 156743.2 | ... | 73053.56 | 3739645 | 346716.8 | 3739755 | 346762.4 |
| 4 | 373.4871 | 3737871 | 349840.7 | 302.8 | 454.6902 | 173991.9 | 153443.2 | | 71573.16 | 3741917 | 343561.9 | 3741774 | 343499.9 |
| 5 | 373.4871 | 3739849 | 346608.7 | 302.3 | 454.6902 | 170804.3 | 150270.6 | | 70259.61 | 3743840 | 340294.9 | 3743797 | 340329.1 |
| 6 | 370.4004 | 3741836 | 343399.8 | 300.8 | 454.6902 | 167665.3 | 147150.7 | | 69113.25 | 3745630 | 337036.6 | 3745744 | 337113.7 |

Clustered route data in UTM coordinates are ready to be run in a neural network. Variables with the blue fill are predictor variables, variables in green are dependent variables.

Figure 9. Clustered Route Data in UTM Format

Figure 9 shows the predictor variables that we use in our models, which we find to produce the best results. Latitudes and longitudes are transformed to UTM coordinates (eastings and northings) as previously described. We demonstrate the accuracy of a neural network for predicting at $\Delta t = 20, 45, \text{and } 120$ minutes into the future from a current vessel location. Through experimentation, we find that using a uniform field of known reference points in the area and then calculating the haversine distance from each point to the location from which we intend to predict from allows us to produce better predictions. These predictions reduce the residual error on the validation set, and therefore also reduce the area of the resulting prediction region. Additionally, we include initial predictor variables Lat.hat and Long.hat calculated as:

$$Lat.hat = Lat + \Delta t \cdot Speed \cdot \cos(\pi \cdot Course / 180) \ , \tag{1.18}$$

and

$$Long.hat = Long + \Delta t \cdot Speed \cdot \sin(\pi \cdot Course / 180) \ . \tag{1.19}$$

Figure 10 shows the clustered route passing through the field of the known 100 reference points.

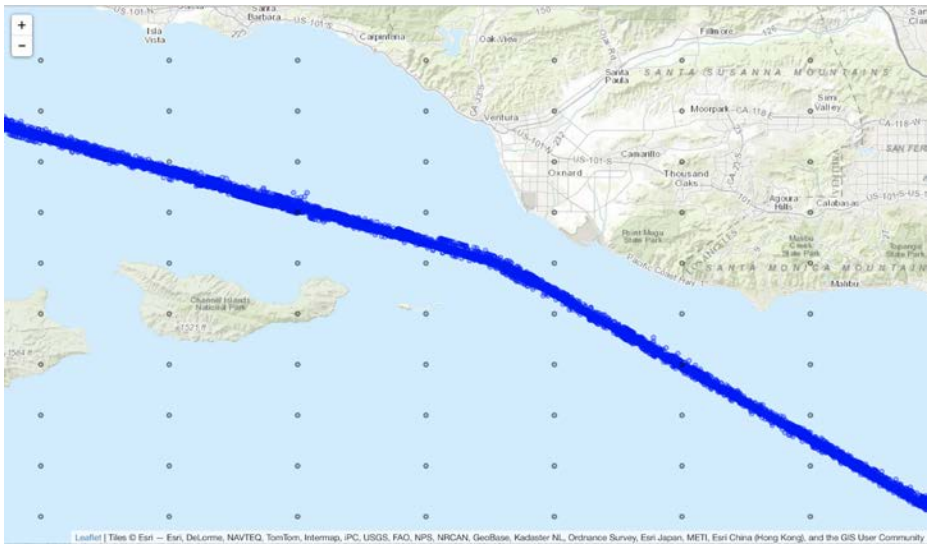



Figure 10. Clustered Route through Field of Known Reference Points

We use the package “H2O” in R (H2O.ai team, 2017) because it provides many options for tuning and because the package allows the user to easily implement parallel computing. The route data above are divided into a training set, a validation set, and a test set, and they are stratified by sub-track. We set aside 10 percent of the sub-tracks to be used as a test set. The remaining sub-tracks are then divided with 80 percent being allocated to the training set, and 20 percent to the validation set. Next, we predict future interpolated northings and future interpolated eastings using a separate model for each. H2O only allows for one dependent variable as is the case for other neural network software Therefore, independent prediction is necessary. Figure 11 depicts an example of how to implement the model using H2O in R.

```
#first predict Northing at t time units ahead
model = h2o.deeplearning(x= c(1:4,6:106),
                         y= "FutLat",
                         training_frame = train,
                         validation_frame = valid,
                         activation = "Rectifier",
                         autoencoder = F,
                         nfolds = 5,
                         hidden = rep(150,3),
                         ignore_const_cols = F,
                         epochs =1000)

#Next predict Long at t time units ahead
model2 = h2o.deeplearning(x= c(1:4,6:106),
                          y= "FutLong",
                          training_frame = train,
                          validation_frame = valid,
                          activation = "Rectifier",
                          autoencoder = F,
                          nfolds = 5,
                          hidden = rep(150,3),
                          ignore_const_cols = F,
                          epochs = 1000)
```

Figure 11. Neural Network Implementation Code

Figure 11 also shows two separate H2O neural network models implemented in R. The function h2o.deeplearning builds a neural network, x refers to the predictor variable columns, and y refers to the dependent variable. In the first model, we are predicting FutLat which represents northings since we have transformed the data into UTM format. We assign the training data frame and validation data frame then choose an activation function. We choose to use "rectifier," the rectified linear activation function, because we find that it works the best for this data based upon the performance of the model in predicting the validation set. We also choose to use cross-fold validation with 5 folds to prevent overfitting to the training set. The $n$ folds are randomly assigned using the nfolds parameter. We also choose three hidden layers, each with 150 nodes designated by the hidden parameter. Finally, we choose to use 1,000 epochs, the number of times the data is cycled through the model to adjust the weights of each connection.

Once the model has been estimated, we predict the future location for each observation in the entire validation set at a specified time in the future. We choose three time periods in this thesis: 20 minutes, 45 minutes, and 120 minutes to evaluate performance. We predict using the R code in Figure 12.

```
Lat.hat = h2o.predict(model,valid,
                      type = "response")[,1]

Long.hat = h2o.predict(model2,valid,
                       type = "response")[,1]
```

Figure 12. Predicting the Validation Set with a Neural Network

We then use quantiles from the residuals of the prediction on the validation set to derive the 95 percent prediction interval on the test set. To estimate the quantiles at the $\alpha = .05$ level of significance, given that we are using two independent models, we use the Bonferroni correction. The Bonferroni correction is widely used when making multiple comparisons or testing multiple hypotheses to adjust for the fact that the likelihood of incorrectly rejecting the null hypothesis increases if the same level of $\alpha$ is used for each test that is required for the overall confidence level. In the context of this thesis, incorrectly rejecting the null hypothesis means incorrectly classifying a vessel as being anomalous based upon a prediction region that is too narrow. To implement the Bonferroni correction, we let $m$ equal the number of hypotheses being tested, then the corrected level of significance for each test is $a/m$ for a one-sided test. This means that we do not estimate the quantiles for which the probability of the dependent variable values being higher or lower is $\alpha/2$ (two-sided interval). Rather, we estimate quantiles for which the probability that the dependent variable being higher or lower is $(\alpha/2)/2$ because we are using two models to form a prediction region. This is accomplished using the R code in Figure 13.

```
lower.lat= quantile(lat.errors,.0125)
upper.lat= quantile(lat.errors,.9875)
```

Figure 13. Estimation of Prediction Error Quantiles in R on Validation Set

In the last step, we predict the northings and eastings for all observations in test set at the chosen time in the future, then add the upper prediction and lower prediction error estimate obtained from the validation set in the previous step to obtain a prediction region. We investigate the performance of this estimation procedure in Chapter IV.

## H. IMPLEMENTATION OF THE RANDOM FOREST

The setup for the random forest model is much the same as for the neural network. We implement the model using the H2O package using the h2o.randomForest function in R as depicted in Figure 14.

```
 98  library(h2o)
 99  localH20 = h2o.init(nthreads= -1)
100  set.seed(4118)
101  train = as.h2o(train)
102  valid = as.h2o(valid)
103  test= as.h2o(test)
104
105  model3=h2o.randomForest(x= c(1:4,6:106, 109:110),
106                          y= "FutLat",
107                          training_frame = train,
108                          validation_frame = valid,
109                          nfolds = 10,
110                          ntrees = ntrees,
111                          max_depth = 20,
112                          score_each_iteration = F,
113                          sample_rate = 1)
```

Figure 14. Random Forest Implementation Code

We implement two models, one to predict northing (FutLat) and one to predict easting (FutLong). We designate that the training data frame is used to create the model through the training_frame = train command. We choose to use 150 independent trees using the ntrees parameter. This number is chosen based on time available to train the models on a laptop computer and because when we used 1,000 trees, the results did not improve significantly on the validation set. We also use cross validation with 10 folds, using the default method for H2O which is to randomly select 10 percent (based on 10 folds) of the validation set observations and assign them to a fold. As the algorithm runs, the predictions made on the validation set are compared to the true future locations and the model is adjusted to account for error. This procedure called cross-validation is repeated for each fold and has been shown to reduce overfitting, and to produce more generalized models. We find that cross validation is effective in our models because the

cross-validated models produce higher accuracy rates when the model is run on an independent test set.

Once the models have been estimated, we use the same methodology for the random forest models as we did for the neural network to calculate the prediction region for future values of northings and eastings.

THIS PAGE INTENTIONALLY LEFT BLANK

# IV. MODEL ANALYSIS AND EVALUATION

Our analysis consists of two parts. First, we present the results of the random forest model predictive performance across three different outgoing clustered routes originating from the ports of Newark, Barcelona, and Los Angeles. Second, for the same routes we present the results of the neural network models. For each route, we assess the prediction accuracy of our models for 20, 45, and 120 minutes into the future.

## A. RANDOM FOREST RESULTS

The random forest models were implemented using the H2O package in R, because H2O allows us to easily parallelize the computation of the random forest model on our computer. All calculations are done on a MacBook-Pro with 2.7GHz quad-core Intel Core i7, and 16GB of memory. H2O allows us to simply perform cross-validation using the validation set, and to parallelize computation. With the long-term goal of automation in mind, we run each model using 150 trees, 10-folds, and maximum tree depth of 20, and a sample rate of 1 (from the predictor columns randomly chosen by the algorithm). The overall results for the random forest are shown in Table 1.

| Random Forest Results | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Location | Prediction time forward (minutes) | Number of sub-tracks (total) | Number of sub-tracks in training set | Number of sub-tracks in validation set | Number of sub-tracks in test set | number of observations in test set | Number of observations contained in 95% interval | % Observations contained by the prediction interval | Lower-bound distance latitude (m) | Upper-bound distance latitude (m) | Lower-bound distance longitude (m) | Upper-bound distance longitude (m) | Average accuracy by prediction Time | Random forest overall average prediction accuracy |
| Newark | 20 | 106 | 76 | 20 | 10 | 1,036 | 953 | 91.99% | -1,025 | 1,520 | -1,431 | 1,384 | 94.43% | 94.24% |
| L.A. | 20 | 139 | 100 | 26 | 13 | 1,583 | 1,573 | 99.37% | -2,232 | 1,545 | -5,209 | 1,960 | | |
| Barcelona | 20 | 170 | 122 | 31 | 17 | 1,135 | 1,019 | 89.78% | -3,030 | 1,485 | -773 | 826 | | |
| Newark | 45 | 106 | 76 | 20 | 10 | 964 | 912 | 94.61% | -1,564 | 2,118 | -2,663 | 3,573 | 96.93% | |
| L.A. | 45 | 139 | 100 | 26 | 13 | 1,530 | 1,526 | 99.74% | -2,260 | 2,107 | -2,340 | 2,861 | | |
| Barcelona | 45 | 170 | 122 | 31 | 17 | 995 | 944 | 94.87% | -1,519 | 1,937 | -1,223 | 1,602 | | |
| Newark | 120 | 106 | 76 | 20 | 10 | 910 | 886 | 97.36% | -2,187 | 1,976 | -11,351 | 10,549 | 90.72% | |
| L.A. | 120 | 139 | 100 | 26 | 13 | 1,193 | 1,066 | 89.35% | -2,964 | 2,728 | -5,737 | 8,476 | | |
| Barcelona | 120 | 170 | 122 | 31 | 17 | 785 | 668 | 85.10% | -3,051 | 3,104 | -2,152 | 1,889 | | |

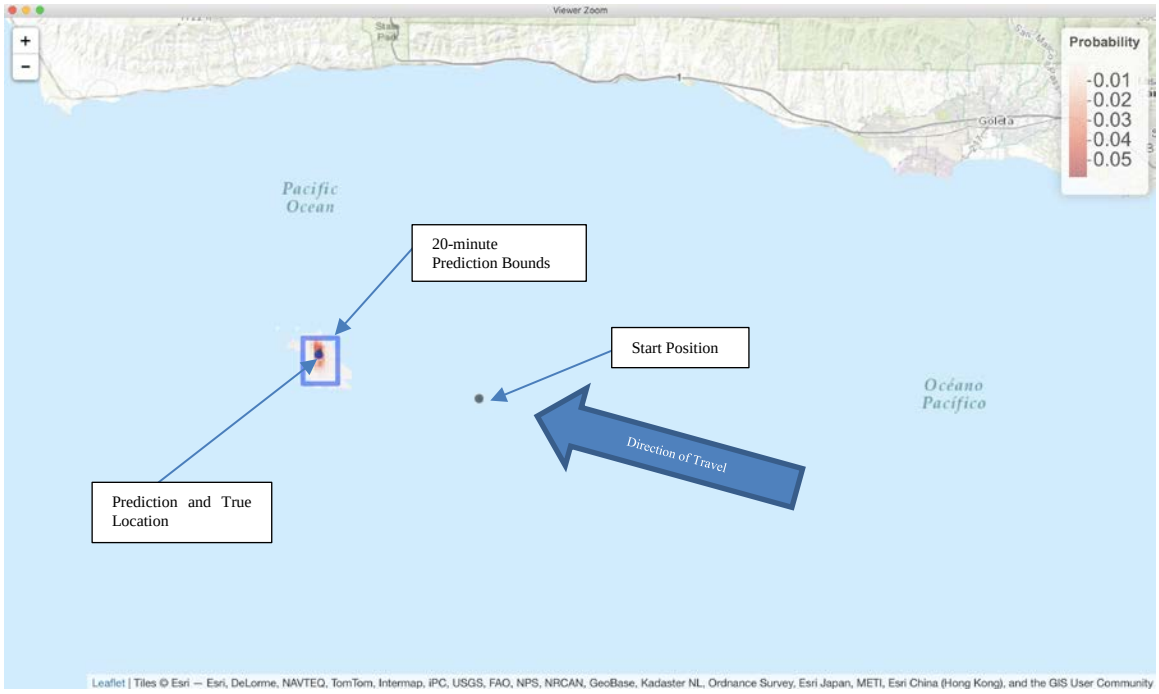
Prediction Results for each of the three regions, and for each time interval.

Table 1. Summary of Random Forest Results

Our twenty-minute prediction accuracy is a 94.43 percent true future vessel locations captured within the predicted latitude and longitude bounds (averaged over the three regions). For the forty-five minute prediction and the two-hour prediction, the average accuracy rates are 96.93 percent, and 90.72 percent respectively. On average the true future position is contained by our prediction interval 94.24 percent of the time across all time intervals and all regions; very close to the expected accuracy rate of 95 percent.

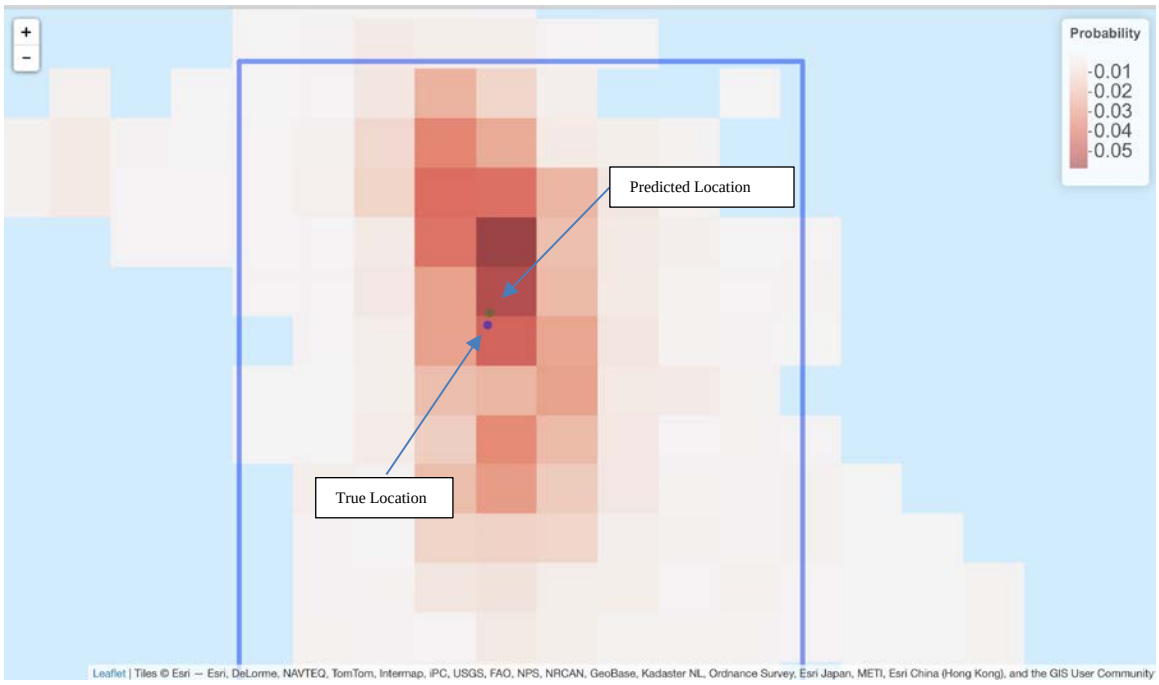
Although the prediction algorithm appears to achieve its objective, it is important to note that the results vary based on location for each of the prediction times depicted in Table 1. We believe that the magnitude of variation seen between the separate locations is due to the small sample size of different vessel sub-tracks. While there are several thousand observations in the validation sets, these observations are correlated within each individual sub-track created as one vessel traverses the route. If a vessel sub-track veers significantly from the others in the route, and the number of sub-tracks in the route is statistically small, it will impose an influence on the bounds of the prediction obtained from the residuals of the validation set. We also note that if the number of sub-tracks in the validation set is small, then the true variance of the route in either direction may not have been fully observed. If a normal sub-track that deviates significantly from all others is predicted, and the validation set also does adequately capture the true variance of the route due to a small number of sub-tracks, then it will not be contained within the estimated bounds and will be flagged erroneously as anomalous.

We now examine the results of the random forest model by first looking at one twenty-minute prediction at a high level in Figure 15, and a closer zoom-in shown in Figure 16.




One twenty-minute prediction off the coast of Los-Angeles. This observation was randomly sampled from the test set. This heatmap was produced using the "raster" package in R (Hijmans, 2016).

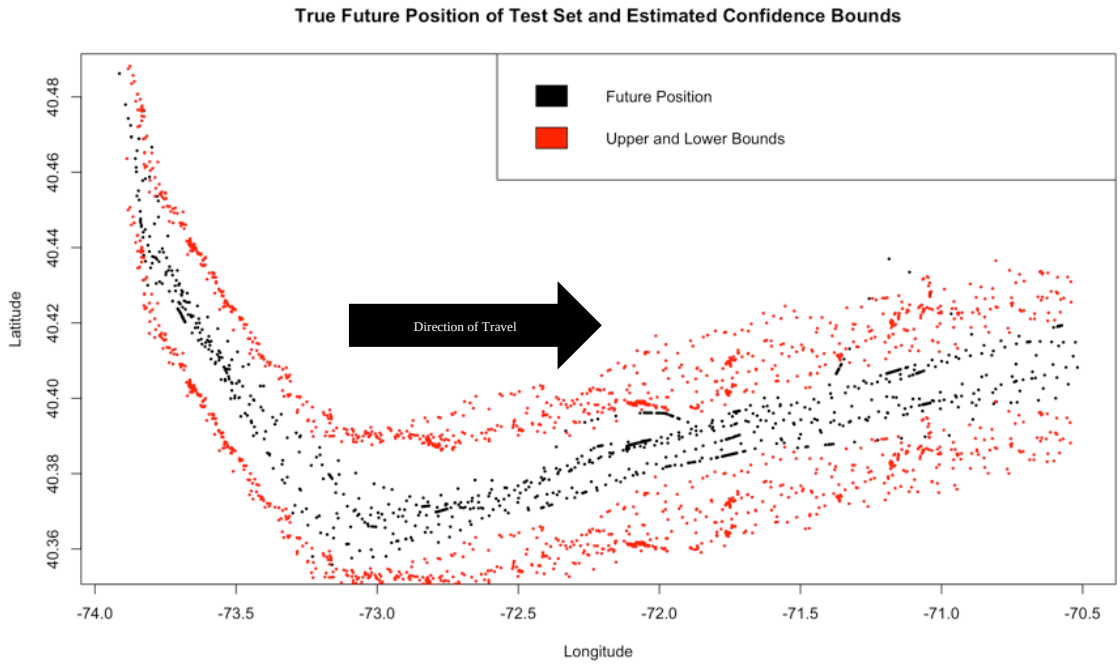
Figure 15. Heatmap of a 20-minute Prediction




This figure shows the same randomly selected test set position prediction as the Figure 15, but at a closer zoom to show how close it was to the true position.

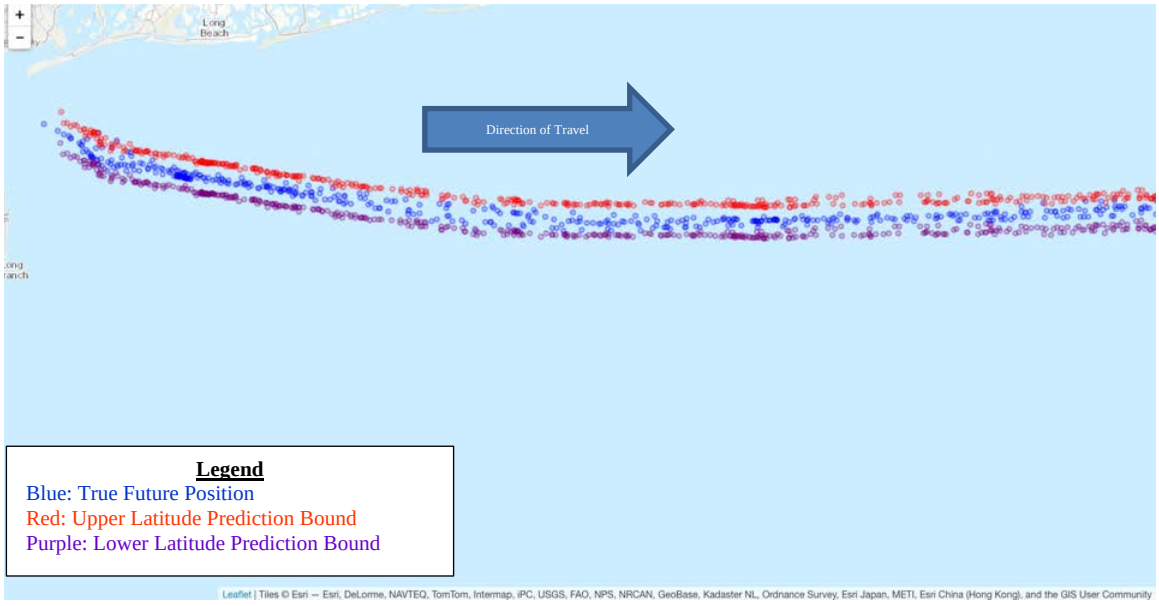
Figure 16. Close View of 20-minute Prediction and True Future Location

Figures 15 and 16 also show how each individual colored box can be prioritized by probability to conduct an efficient search mission. One might start searching boxes with a higher probability, then move to boxes with lower priority in an efficient manner. The distribution shown in Figures 15 and 16 is derived from the validation prediction errors for the entire route. The distribution is for the whole of the route and does not change depending upon the location of the prediction. Therefore, the predicted position is not located in the box with the highest probability because the test set has a slightly different distribution. As the sample size of the sub-tracks contained in the route increases, we expect the distributions of the validation errors to look much the same as the distribution of the test set error and for predictions to fall within the box containing the highest probability. Figures 17 and 18 depict a two-hour prediction for every observation of every outgoing (East-Heading) sub-track in the test set predicted off the coast of New York City. These vessels have departed the Port of Newark-Elizabeth and are heading East toward an intersection (POI) of routes.




The black dots represent the true position of every observation of every vessel in the test set two hours into the future. The red points are the associated bounds for Latitude. We exclude the Longitude bounds for presentation purposes. The curvature appears to be greater than it is due to scaling.

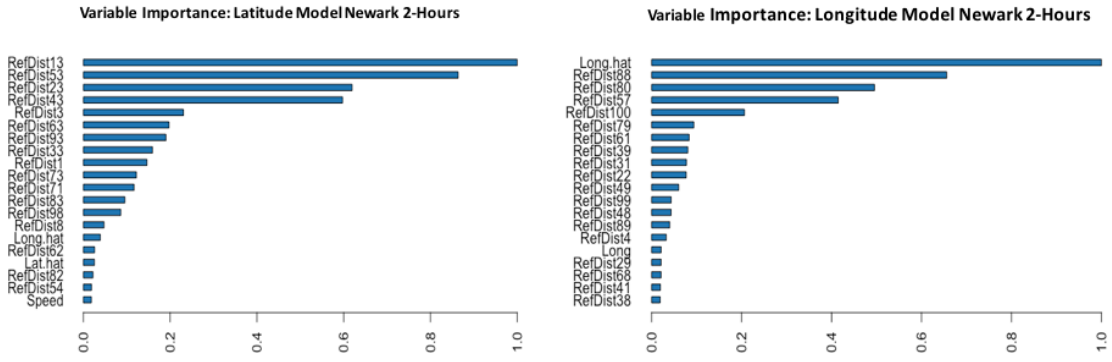
Figure 17. Two-hour Predictions Port of Newark-Elizabeth




The same future positions and Latitude bounds from Figure 17, but overlaid on a map. In this figure, the true positions are in blue, and the upper and lower latitude bounds are depicted in red and purple.

Figure 18. Two-hour Predictions Port of Newark-Elizabeth (Map Version)

As mentioned in Chapter III, the variables used in the model are Speed, Latitude (Northing), Longitude(Easting), Course, Average Speed, Reference-Point Distances (1 through 100), Lat Hat, and Long Hat. We briefly summarize the top twenty variables by plotting the importance of each (a feature of the H2O package). Figure 19 shows the importance of the reference distances in our models. We believe that these distances generated from uniformly distributed points throughout the route area allow our model to capture curvature which can be thought of as interaction between latitude and longitude along the route. They are important because they allow the model to produce better predictions than if they are not present in the model. This is true for each route and each region under consideration. Figure 19 also shows that the naïve prediction (Long.hat) is the most important in predicting Longitude when departing the port of Newark, but it also shows that Lat.hat is not useful in predicting latitude for a two-hour prediction. The random forest uses the variables that matter the most for determining a prediction, and, because of the curvature of the Newark route and the length of the prediction time Lat.hat does not contribute to this model. In contrast, for a twenty-minute prediction on the same route, Lat.hat is the third most significant variable in prediction the route. This shows the flexibility of the random forest to discriminate between variables that matter the most based on the spatial characteristics of the route and the time period of the prediction.



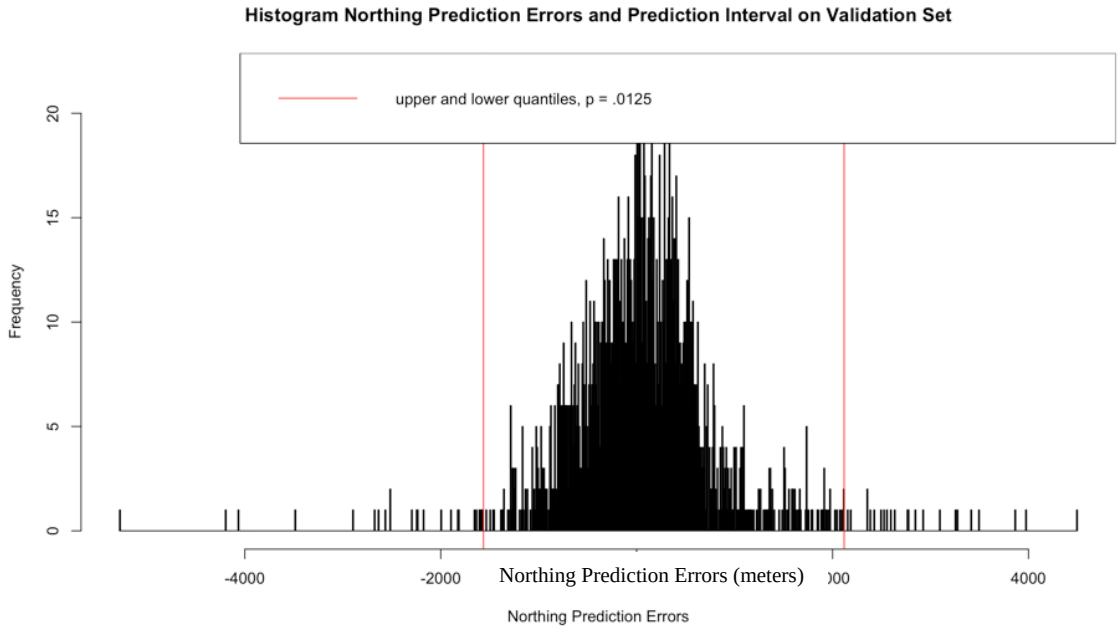

The top twenty predictors for a two-hour prediction. Variables are ordered in order of high importance to lower importance from top to bottom.

Figure 19. Random Forest Variable Importance

We now describe the process of obtaining predictions. As previously mentioned, the residuals are calculated by taking the difference between the predicted position and the actual position:

$$y_{valid\,(Northing)} - \hat{y}_{valid\,(Northing)} \text{ and } y_{valid\,(Easting)} - \hat{y}_{valid\,(Easting)}. \tag{1.20}$$

It is critical to first transform the coordinate system from latitude and longitude to UTM, which is a representation of the earth that approximates the earth by a series of flat surfaces (like a disco ball). This is important because it transforms the units from angles to meters that can then be used to determine the magnitude of the error. Once the residuals are calculated in meters, the quantiles may be extracted as depicted in Figures 20 and 21.




A histogram of the residuals calculated for the Northing errors from the predictions of every observation in the validation set. Quantiles are calculated and extracted using the quantile function in R. The quantiles are depicted by the red vertical lines here (roughly 2,000 meters in either direction of the prediction). These quantiles are then added to the test set predictions to estimate the prediction bounds.

Figure 20. Northing Residuals and Quantiles (Validation Set)

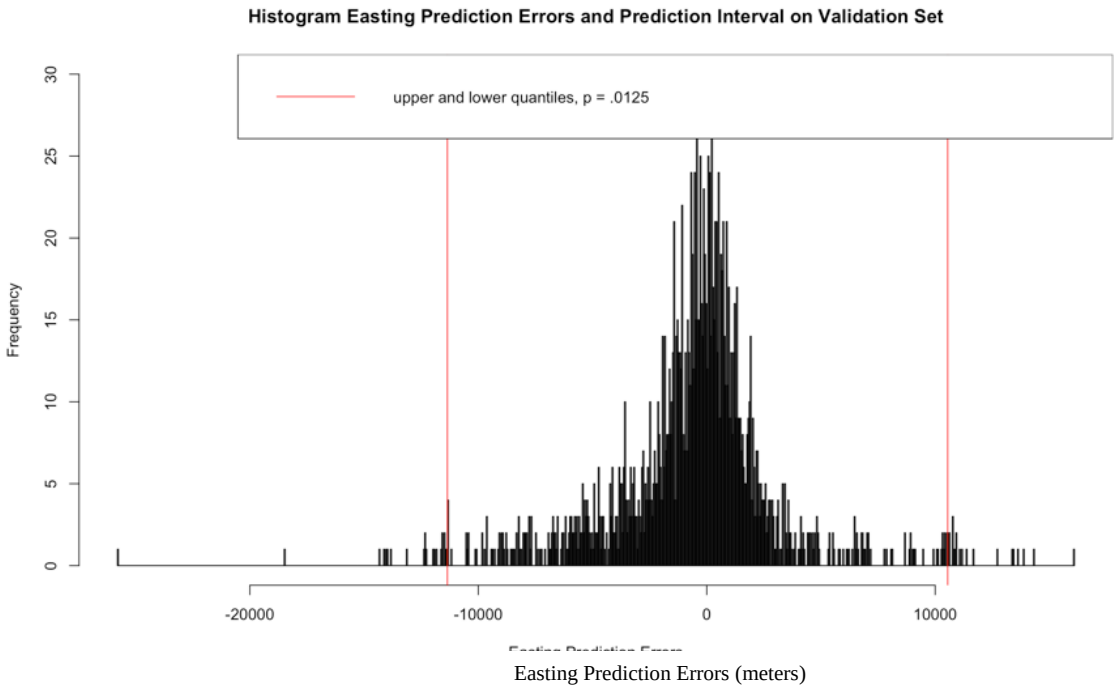



Figure 21. Easting Residuals and Quantiles (Validation Set)

## B. NEURAL NETWORK RESULTS

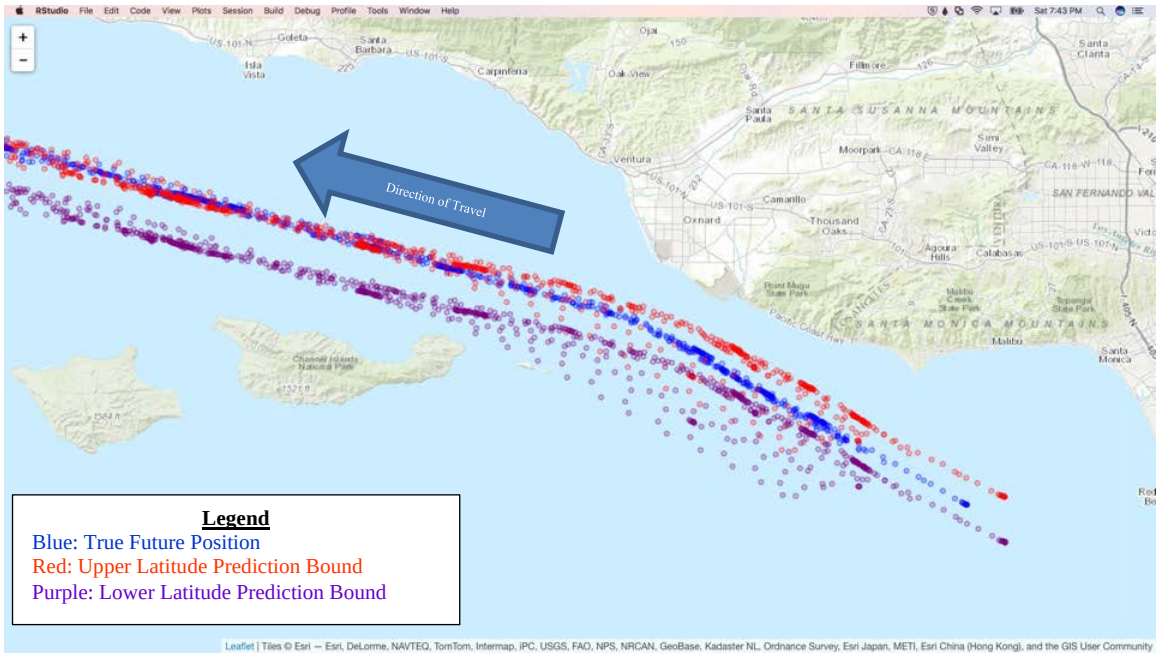
We use the same methodology using the neural network to predict the outcomes. However, we do not use the naïve prediction (Lat.hat and Long.hat) of future position because it diminishes the performance of the model on the validation set. As mentioned in Chapter III, the same procedure is used to estimate the prediction bounds as we use for the random forest model.

As Table 2 suggests, the neural network model performance does not compare favorably to that obtained with the random forest model.

| Neural Network Results | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Location | Prediction time forward (minutes) | Number of sub-tracks (total) | Number of sub-tracks in training set | Number of sub-tracks in validation set | Number of sub-tracks in test set | number of observations in test set | Number of observations contained in 95% interval | % Observations contained by the prediction interval | Lower-bound distance latitude (m) | Upper-bound distance latitude (m) | Lower-bound distance longitude (m) | Upper-bound distance longitude (m) | Average accuracy by prediction Time | Neural network overall average prediction accuracy |
| Newark | 20 | 106 | 76 | 20 | 10 | 1,064 | 381 | 35.81% | -1,099 | 1,221 | -1,314 | 8,898 | 80.66% | 81.19% |
| L.A. | 20 | 139 | 100 | 26 | 13 | 1,957 | 1,855 | 94.79% | -561 | 648 | -722 | 808 | | |
| Barcelona | 20 | 170 | 122 | 31 | 17 | 1,110 | 1,096 | 98.74% | -1,600 | 935 | -1,241 | 1,426 | | |
| Newark | 45 | 106 | 76 | 20 | 10 | 965 | 790 | 81.87% | -3,416 | 1,518 | -3,005 | 4,085 | 83.07% | |
| L.A. | 45 | 139 | 100 | 26 | 13 | 1,516 | 1,122 | 74.01% | -1,655 | 1,815 | -6,163 | 1,582 | | |
| Barcelona | 45 | 170 | 122 | 31 | 17 | 1,058 | 1,028 | 97.16% | -3,150 | 1,312 | -2,058 | 4,912 | | |
| Newark | 120 | 106 | 76 | 20 | 10 | 1,042 | 803 | 77.06% | -2,639 | 1,668 | -8,693 | 15,227 | 79.66% | |
| L.A. | 120 | 139 | 100 | 26 | 13 | 1,014 | 750 | 73.96% | -5,675 | 989 | -9,885 | 8,790 | | |
| Barcelona | 120 | 170 | 122 | 31 | 17 | 894 | 797 | 89.15% | -3,718 | 4,434 | -2,064 | 3,098 | | |

Table 2. Summary of Neural Network Results

The results obtained are nearly 15 percent lower on average overall. The variability within each period is also greater. For example, in Newark a twenty-minute prediction interval only contained the true position 35 percent of the time. The process of finding the optimal parameter combination using neural networks is done by trial and error and requires considerable time. A neural network likely requires many more epochs than the 1,000 that we used, although we did run iterations with 10,000 epochs and did not obtain results that were significantly better. We observe that neural networks are not as effective as random forests for capturing the curvature of a route which is reflected in the magnitude of the residuals and the prediction bounds. An example of this behavior is depicted in Figure 22.




Note the true positions in the top left region of the route are not well captured by the prediction bounds. Additionally, the lower latitude bounds (purple) are clearly being distorted by the curvature of the route.

Figure 22. Neural Network Two-hour Prediction off the Coast of Los Angeles

Figure 22 can be compared to an independent prediction by the random forest model of the same route in Figure 23.

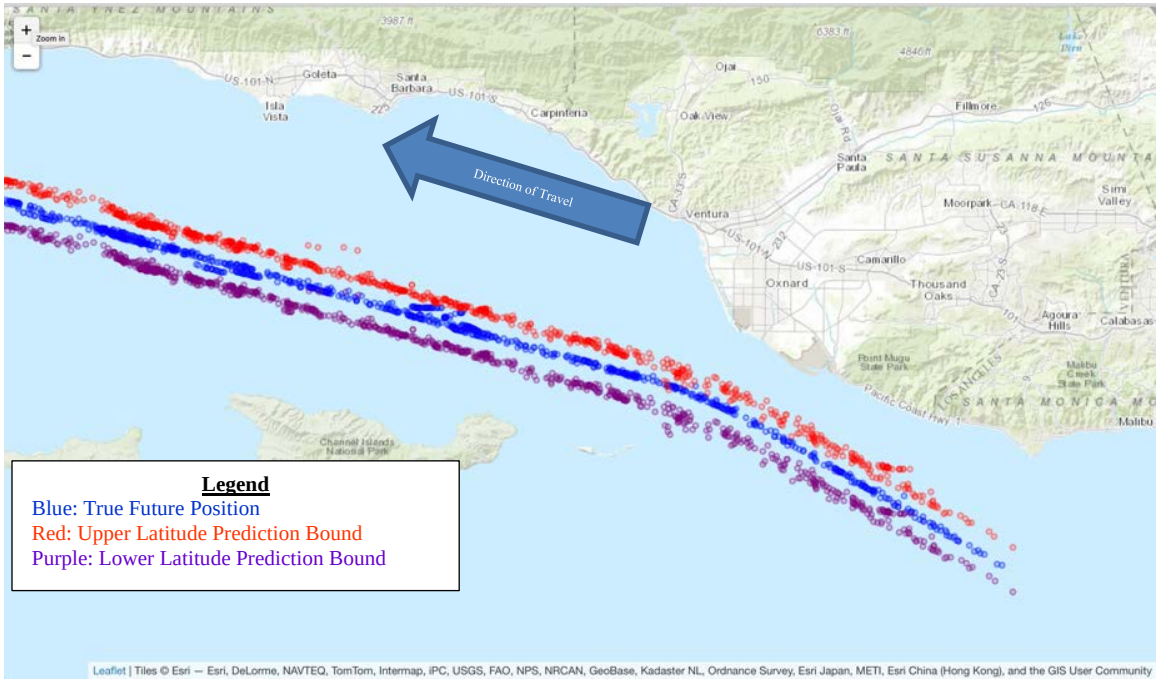



Figure 23. Random Forest Comparison Two-hour Prediction off the Coast of Los Angeles

Our results show that a neural network is less effective at capturing the curvature of the route, and takes longer to run on our laptop computer. A neural network also shows less promise for automated use across different regions because each model for every new route would need to be tuned specifically to that route.

THIS PAGE INTENTIONALLY LEFT BLANK

# V. SUMMARY AND RECOMMENDATIONS

## A. SUMMARY

In this thesis, we present two separate models to predict future vessel locations along a clustered route in three different regions. The random forest outperforms the neural network in all regions. The prediction intervals produced by the random forest model achieve 94 percent containment of the true future position, which is close to the targeted 95 percent containment. The prediction intervals obtained from the neural network only contain the true future positions 81 percent of the time on average and produce more variability within each time group. Random forest models have proven to be both simpler to implement and faster to run. The random forest models capture the curvature of the routes connecting a network of different POIs consisting of ports, and route intersections, and our implementation of the neural network does not capture this curvature well.

We find that linear estimates (naïve prediction) of the future latitude position and longitude prediction are useful as predictor variables in a random forest model. These variables were almost always ranked among the top 20 variables in each of the models run across different prediction time frames in three locations around the world. These variables may be of less importance as the prediction time-frame increases beyond the durations that were covered in this thesis.

We find that using the haversine distance from 100 evenly distributed reference points across the extent of the geographic area covered by the route under consideration decreases the variance of the residual error of the prediction on a validation set. This, we hypothesize, may be attributed to a partial capture of an interaction between latitude and longitude values within the route.

We have shown that a heatmap within the prediction region can be used to efficiently allocate resources in a search and rescue scenario. Assets should be allocated to boxes within the prediction region that have a higher probability of containing the vessel at a specified time in the future.

We find that random forest models implemented on our laptop are more accurate than the neural network models, but this finding may not always hold. It is possible that with more data, and future development of automatic neural network tuning, that a neural network could deliver results that compare favorably to a random forest model, and be implemented automatically.

## B. RECOMMENDATIONS

This work focuses on obtaining reliable predictions between POIs consisting of ports and intersections of routes, along with the associated uncertainty bounds of those predictions. Future work should seek to obtain more data and show that as the number of sub-tracks in the predicted route increases, the bounds calculated by the quantile estimation process of the validation set converges to 95 percent or better.

The methodology set forth in this thesis needs to be integrated into a networked prediction model which accounts for predictions that go beyond an intersection of routes. Other authors have clustered Points of Interest (POIs) consisting of ports, route intersections, and turning points. A network within a geographic region can be attained by clustering these points and then deriving the vessel sub-tracks that connect the POIs. The links between these POIs will not always be a straight line and would preclude the use of a linear model to predict vessel location along these curved links. Our methodology might be used to predict along these curved links and combined with other methods to predict beyond an intersection of routes.

The data filtering process should be improved to run more efficiently. We use an extensive process to clean the data before running our models. Our algorithms, which are implemented in R, can be reproduced in another language such as Python for faster run-times and the algorithms themselves may also be improved for efficiency. Finally, this methodology might be applied to three-dimensional fields such as aviation or undersea warfare in the future and could prove valuable in the process of deconflicting airspace.

# APPENDIX. AIS DATA DICTIONARY

| Parameter | Bits | Description |
|---|---|---|
| Message ID | 6 | Identifier for this message 1, 2 or 3 |
| Repeat indicator | 2 | Used by the repeater to indicate how many times a message has been repeated. See Section 4.6.1, Annex 2; 0-3; 0 = default; 3 = do not repeat any more |
| User ID | 30 | MMSI number |
| Navigational status | 4 | 0 = under way using engine, 1 = at anchor, 2 = not under command, 3 = restricted maneuverability, 4 = constrained by her draught, 5 = moored, 6 = aground, 7 = engaged in fishing, 8 = under way sailing, 9 = reserved for future amendment of navigational status for ships carrying DG, HS, or MP, or IMO hazard or pollutant category C, high speed craft (HSC), 10 = reserved for future amendment of navigational status for ships carrying dangerous goods (DG), harmful substances (HS) or marine pollutants (MP), or IMO hazard or pollutant category A, wing in ground (WIG); 11 = power-driven vessel towing astern (regional use); 12 = power-driven vessel pushing ahead or towing alongside (regional use);<br>13 = reserved for future use,<br>14 = AIS-SART (active), MOB-AIS, EPIRB-AIS<br>15 = undefined = default (also used by AIS-SART, MOB-AIS and EPIRB-AIS under test) |
| Rate of turn $ROT_{AIS}$ | 8 | 0 to +126 = turning right at up to 708 deg per min or higher<br>0 to -126 = turning left at up to 708 deg per min or higher Values between 0 and 708 deg per min coded by $ROT_{AIS} = 4.733\ SQRT(ROT_{sensor})$ degrees per min<br>where $ROT_{sensor}$ is the Rate of Turn as input by an external Rate of Turn Indicator (TI). $ROT_{AIS}$ is rounded to the nearest integer value.<br>+127 = turning right at more than 5 deg per 30 s (No TI available)<br>-127 = turning left at more than 5 deg per 30 s (No TI available)<br>-128 (80 hex) indicates no turn information available (default).<br>ROT data should not be derived from COG information. |
| SOG | 10 | Speed over ground in 1/10 knot steps (0-102.2 knots)<br>1 023 = not available, 1 022 = 102.2 knots or higher |
| Position accuracy | 1 | The position accuracy (PA) flag should be determined in accordance with the table below:<br>1 = high (<= 10 m)<br>0 = low (> 10 m)<br>0 = default |
| Longitude | 28 | Longitude in 1/10 000 min (+/-180 deg, East = positive (as per 2's complement), West = negative (as per 2's complement).<br>181= (6791AC0h) = not available = default) |
| Latitude | 27 | Latitude in 1/10 000 min (+/-90 deg, North = positive (as per 2's complement), South = negative (as per 2's complement). 91deg (3412140h) = not available = default) |
| COG | 12 | Course over ground in 1/10 = (0-3599). 3600 (E10h) = not available = default. 3 601-4 095 should not be used |
| True heading | 9 | Degrees (0-359) (511 indicates not available = default) |
| Time stamp | 6 | UTC second when the report was generated by the electronic position system (EPFS) (0-59, or 60 if time stamp is not available, which should also be the default value, or 61 if positioning system is in manual input mode, or 62 if electronic position fixing system operates in estimated (dead reckoning) mode, or 63 if the positioning system is inoperative) |
| special maneuvre indicator | 2 | 0 = not available = default<br>1 = not engaged in special maneuver<br>2 = engaged in special maneuver<br>(i.e.: regional passing arrangement on Inland Waterway) |
| Spare | 3 | Not used. Should be set to zero. Reserved for future use. |
| RAIM-flag | 1 | Receiver autonomous integrity monitoring (RAIM) flag of electronic position fixing device; 0 = RAIM not in use = default; 1 = RAIM in use. See Table |
| Communication state (see below) | 19 | See Rec. ITU-R M.1371-5 Table 49 |
| Number of bits | 168 | |

Table 3. AIS Dynamic Data Dictionary. Source: USCG (2017).

| Parameter | Bits | Description |
|---|---|---|
| Message ID | 6 | Identifier for this Message |
| Repeat indicator | 2 | Used by the repeater to indicate how many times a message has been repeated.<br>Refer to §?4.6.1, Annex 2; 0-3; 0 = default; 3 = do not repeat any more |
| User ID | 30 | MMSI number |
| AIS version indicator | 2 | 0 = station compliant with Recommendation ITU-R M.1371-1<br>1 = station compliant with Recommendation ITU-R M.1371-3 (or later)<br>2 = station compliant with Recommendation ITU-R M.1371-5 (or later)<br>3 = station compliant with future editions |
| IMO number | 30 | 0 = not available = default – Not applicable to SAR aircraft<br>0000000001-0000999999 not used<br>0001000000-0009999999 = valid IMO number;<br>0010000000-1073741823 = official flag state number. |
| Call sign | 42 | 7?=?6 bit ASCII characters, @@@@@@@ = not available = default<br>Craft associated with a parent vessel, should use "A" followed by the last 6 digits of the MMSI of the parent vessel. Examples of these craft include towed vessels, rescue boats, tenders, lifeboats and liferafts. |
| Name | 120 | Maximum 20 characters 6 bit ASCII<br>"@@@@@@@@@@@@@@@@@@@@" = not available = default<br>The Name should be as shown on the station radio license. For SAR aircraft, it should be set to "SAR AIRCRAFT NNNNNNN" where NNNNNNN equals the aircraft registration number. |
| Type of ship and cargo type | 8 | 0 = not available or no ship = default<br>1-99 = as defined below<br>100-199 = reserved, for regional use<br>200-255 = reserved, for future use<br>Not applicable to SAR aircraft |
| Overall dimension/ reference for position | 30 | Reference point for reported position.<br>Also indicates the dimension of ship (m) (see below)<br>For SAR aircraft, the use of this field may be decided by the responsible administration. If used it should indicate the maximum dimensions of the craft. As default should A = B = C = D be set to "0" |
| Type of electronic position fixing device | 4 | 0 = undefined (default)<br>1 = GPS<br>2 = GLONASS<br>3 = combined GPS/GLONASS<br>4 = Loran-C<br>5 = Chayka<br>6 = integrated navigation system<br>7 = surveyed<br>8 = Galileo,<br>9-14 = not used<br>15 = internal GNSS |
| ETA | 20 | Estimated time of arrival; MMDDHHMM UTC<br>Bits 19-16: month; 1-12; 0 = not available = default<br>Bits 15-11: day; 1-31; 0 = not available = default<br>Bits 10-6: hour; 0-23; 24 = not available = default<br>Bits 5-0: minute; 0-59; 60 = not available = default<br>For SAR aircraft, the use of this field may be decided by the responsible administration |
| Maximum present static draught | 8 | In 1/10 m, 255 = draught 25.5 m or greater, 0 = not available = default; in accordance with IMO Resolution A.851<br>Not applicable to SAR aircraft, should be set to 0 |
| Destination | 120 | Maximum 20 characters using 6-bit ASCII;<br>@@@@@@@@@@@@@@@@@@@@ = not available<br>For SAR aircraft, the use of this field may be decided by the responsible administration |
| DTE | 1 | Data terminal equipment (DTE) ready (0 = available, 1 = not available = default) |
| Spare | 1 | Spare. Not used. Should be set to zero. Reserved for future use. |
| Number of bits | 424 | Occupies 2 slots |

Table 4. AIS Static Data Dictionary. Source: USCG (2017).

# LIST OF REFERENCES


Balduzzi, M., Pasta, A., & Wilhoit, K. (2014). A security evaluation of AIS Automated Identification System. *30th Annual Computer Security Applications Conference* (pp. 436–445). doi: 10.1145/2664243.2664257

Bay, S. (2017). *Evaluation of factors on the patterns of ship movement and predictability of future ship location in the Gulf of Mexico.* (Master's thesis). Retrieved from http://calhoun.nps.edu/handle/10945/53021.

Cheng, J., Karambelkar, B., & Xie, Y. (2017). Leaflet: Create Interactive Web Maps with the JavaScript 'Leaflet' Library. R package version 1.1.0. https://CRAN.R-project.org/package=leaflet

Department of the Navy. (2007). Maritime Domain Awareness Concept. Washington, DC: Chief of Naval Operations. Retrieved from http://www.navy.mil/navydata/cno/Navy_Maritime_Domain_Awareness_Concept_FINAL_2007.pdf

The H2O.ai Team. (2017). h2o: R Interface for H2O. R package version 3.10.4.6. https://CRAN.R-project.org/package=h2o

Hampton, B. (2009, March 24). AIS on boats—What is that ship doing? Retrieved from http://yachtpals.com/ais-boats-4116.

Harati-Mokhtari, A., Wall, A., Brooks, P., & Wang, J. (2007). Automatic Identification System (AIS): Data reliability and human error implications. *The Journal of Navigation, 60*(3), 373-389.

Hastie, T., Tibshiran, R., & Friedman, J. (2009). *The elements of statistical learning*. New York: Springer Science + Business Media, LLC.

Hijmans, R. J. (2015). Geosphere: Spherical trigonometry. Retrieved from https://CRAN.R-project.org/package=geosphere.

Hijmans, R.J (2016). raster: Geographic data analysis and modeling. R package version 2.5-8. https://CRAN.R-project.org/package=raster

Kaufman, L., & Rousseeuw, P. J. (1990). *Finding groups in data, an introduction to cluster analysis.* Brussels: John Wiley & Sons, Inc.

Khan, A., Cees, B., & Kaye, M. (2005). Ship motion prediction for launch and recovery of air vehicles. *Proceedings of MTS/IEEE OCEANS*. doi: 10.1109/OCEANS.2005.1640198

Koyak, R. (2017). Statistical tool for the analysis of Automated Information System (AIS) data final report. Unpublished manuscript.

Maechler, M., Rousseeuw, P., Struyf, A., Hubert, M., Hornik, K. (2016). Cluster: Cluster analysis basics and extensions. R package version 2.0.5.

Mao, S., Tu, E., Zhang, G., Rachmawati, L., Rajabally, E., & Huang, G.-B. (2016). *An Automatic Identification System (AIS) database for maritime trajectory prediction and data mining.* Retreived From https://arxiv.org/pdf/1607.03306.pdf

McAbee, A. (2013). *Traffic pattern detection using the Hough transformation for anomaly detection to improve maritime domain awareness.* (Master's thesis). Retrieved from http://calhoun.nps.edu/handle/10945/38977

Millifiori, L., Braca, P., Bryan, K., & Willett, P. (2016). Modeling vessel kinematics using a stochastic mean-reverting process for long-term prediction. *IEEE Transactions on Aerospace and Electronic Systems, 52*(5). doi: 10.1109/TAES.2016.150596

Morris, B., & Trivedi, M. (2008). A survey of vision-based trajectory learning and analysis for surveillance. *IEEE Transactions on Circuits and Systems for Video Technology, 18*(8), 1114–1127. doi: 10.1109/TCSVT.2008.927109

Morris, B., & Trivedi, M. (2011). Trajectory learning for activity understanding: unsupervised, multilevel, and long-term adaptive approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 33*(11), 2287–2301. doi: 10.1109/TPAMI.2011.64

Palacios, R., & Doshi, A. G. (2008). Computing aircraft position prediction. *The Open Transportation Journal, 2*, 94–97. Retrieved from https://benthamopen.com/contents/pdf/TOTJ/TOTJ-2-94.pdf

Pallota, G., Horn, S., Braca, P., & Bryan, K. (2014). Context-enhanced vessel prediction based on Ornstein-Uhlenbeck proceses using historical AIS traffic patterns: Real-world experimental results. *Fusion.* Retrieved from http://ieeexplore.ieee.org/document/6916016/?part=1

Pallotta, G., Vespe, M., & Bryan, K. (2013). Vessel pattern knowledge discovery from AIS data: A framework for anomaly detection and route prediction. *Entropy, 15*(6), 2218-2245. Retrieved from http://www.mdpi.com/1099-4300/15/6/2218

Pebesma, E.J., R.S. Bivand, 2005. Classes and methods for spatial data in R. R News 5 (2), https://cran.r-project.org/doc/Rnews/

R Core Team (2016). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/

Ristic, B., La Scala, B., Morelande, M., & Gordon, N. (2008). Statistical analysis of motion patterns in AIS data: Anomaly detection and motion prediction. *Information Fusion*, 40-46. Retrieved from http://ieeexplore.ieee.org/document/4632190/

Stone, L., Streit, R., Corwin, T., & Bell, K. (2014). *Bayesian multiple target tracking.* Norwood, MA: Artech House.

Tester, K. A. (2013). *A spatiotemporal clustering approach to maritime domain awareness.* (Master's thesis). Retrieved from http://calhoun.nps.edu/handle/10945/37731

Tu, E., Zhang, G., Rachmawati, L., Rajabally, E., & Huang, G.-B. (2016). *Exploiting AIS data for intelligent maritime navigation: A comprehensive survey*. Retrieved from https://arxiv.org/abs/1606.00981

United States Coast Guard Navigation Center. (2017, April 25). Types of automatic identification systems. Retrieved from https://www.navcen.uscg.gov/?pageName=typesAIS

Vespe, M., Visentini, I., Bryan, K., & Braca, P. (2012). Unsupervised learning of maritime traffic patterns for anomaly detection. *Entropy* 2013, 15, 2218–2245. doi:10.3390/e15062218

Zhu, F. (2011). Mining ship trajectory patterns from AIS database for maritime surveillance. *Emergency Management and Management Sciences (ICEMMS)*. doi: 10.1109/ICEMMS.2011.6015796

THIS PAGE INTENTIONALLY LEFT BLANK

# INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California