MARITIME TRANSPORTATION RESEARCH AND EDUCATION CENTER
TIER 1 UNIVERSITY TRANSPORTATION CENTER
U.S. DEPARTMENT OF TRANSPORTATION

Project Title
Novel Big Data and Artificial Intelligence Analytics Methods for Tracking and Monitoring Maritime Traffics

Project start and end dates
October 2022 – December 2023

Principal Investigator(s) and contact information (including institution name)

Robert W. Whalin, Ph.D., Project PI
Professor, Department of Civil and Environmental Engineering, Jackson State University, Jackson, Mississippi 39217
robert.w.whalin@jsums.edu

Tor A. Kwembe, Ph.D., Project Co-PI
Professor, Department of Mathematics and Statistical Sciences, Jackson State University, Jackson, Mississippi 39217
tor.a.kwembe@jsums.edu

Other Author Names
Eric S. Jackson, CDS&E Ph.D. Student
Professor, Department of Mathematics and Statistical Sciences, Jackson State University, Jackson, Mississippi 39217
eric.jackson36@gmail.com

Lancelot Nelson, CDS&E Ph.D. Student
Professor, Department of Mathematics and Statistical Sciences, Jackson State University, Jackson, Mississippi 39217
Lancelot.nelson@students.jsums.edu

Ingrid K. Tchakoua, CDS&E Ph.D. Student
Professor, Department of Mathematics and Statistical Sciences, Jackson State University, Jackson, Mississippi 39217
Ingrid.k.tchakoua@students.jsums.edu

FINAL RESEARCH REPORT
Prepared for:
Maritime Transportation Research and Education Center

ACKNOWLEDGEMENT
This material is based upon work supported by the U.S. Department of Transportation under Grant Award Number 69A3551747130. The work was conducted through the Maritime Transportation Research and Education Center at the University of Arkansas.

DISCLAIMER
The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated in the interest of information exchange. The report is funded, partially or entirely, by a grant from the U.S. Department of Transportation's University Transportation Centers Program. However, the U.S. Government assumes no liability for the contents or use thereof.

Contents

1. Project Description

Maritime transportation represents a substantial portion of international trade. Sustainable development of marine transportation requires systematic modeling and surveillance for maritime situational awareness. Big data and artificial intelligence (AI) are crucial components of data-driven decision-making in most industries. AI is gradually transforming the traditional operational process of the maritime industry. Consequently, the amount of research on the application of big data and AI has increased significantly since 2012. For example, a common method in the evaluation of ship speed involves computing the total resistance of a ship using theoretical analysis; however, using theoretical equations cannot be applied for most ships under various operating conditions. So, machine learning approaches have been proposed to predict ship speed over the ground and in monitoring and tracking vessels.

The past decade has seen an explosion of machine learning research and applications; especially, deep learning methods have enabled key advances in many application domains, such as computer vision, speech processing, and in maritime vessel trajectories. However, the performance of many machine learning methods is very sensitive to a plethora of design decisions, which constitutes a considerable barrier for new users. This is particularly true in the booming field of deep learning, where human engineers need to select the right neural architectures, training procedures, regularization methods, and hyper-parameters of all of these components in order to make their networks do what they are supposed to do with sufficient performance. This process has to be repeated for every application. Even experts are often left with tedious episodes of trial and error until they identify a good set of choices for a particular dataset. The purpose of the field of Automated Machine Learning (AutoML) is to make these decisions in a data-driven, objective, and automated way. That is, the user simply provides data, and the AutoML system automatically determines the approach that performs best for this particular application.

Therefore, the purpose of this project is to develop an AutoML model for monitoring and tracking maritime traffic, a state-of-the-art machine learning approach that is accessible to users of maritime historical and real-time databases interested in applying machine learning but do not have the resources to learn about the methods and technologies behind it in detail. This can be seen as a democratization of machine learning where the state-of-the-art machine learning is at every maritime database user's fingertip.

Since the configuration of the global maritime network is organized along a circum-equatorial corridor linking North America, Europe, and Pacific Asia through the Suez Canal, the Strait of Malacca, and the Panama Canal (that is, linking all the choke points). The Machine Learning and hence the AutoML models are modified to capture maritime traffic in all global waters, including inland vessel traffics equip with the AIS transponders.

The project has engaged three CDS&E Ph.D. students with excellent academic records in a yearlong research activity in the use of the Automatic Identification System (AIS) datasets to develop maritime traffic tracking and monitoring models. Consequently, there will be three Ph.D. dissertations resulting from the outcomes of the project. The program Graduate Assistants will acquire broad data science and big data analytics skills geared towards effective applications of the CDS&E methods in novel maritime traffic modeling and analysis. There will be three scholarly articles resulting from the project. The plan is to publish them in peer reviewed Transportation professional journals.

2. Methodological Approach

The methodology consists of the details of the AIS historical data acquisition from Spire, a proper pre-processing technique, and feature selection for the acquired dataset. We have also provided details of the development and implementation of the various vessel tracking and monitoring models utilizing the different variations of machine learning methodologies, the optimization of hyperparameter of the selected models, a comparison of the models to determine the most efficient modelling method, and finally putting them all together to build and Automated Machine Learning (Auto ML) end-to-end algorithm. Therefore, the methodology is broken down into two major phases: Data-related activities and machine learning methods for maritime data analytics and vessel tracking activities. Data activities include data collection and data preparation. The machine learning methods for maritime data analysis includes the development and selection of models that are specific for historical and real-time maritime datasets and results. A schematic depiction of the developed approaches is given in Figure 1 and Figure 2.

Data Acquisition and Source

A Global AIS satellite, terrestrial, and dynamic historical data was acquired from Spire [cite] for the time period of June 26, 2022 to July 10, 2022. The data collected started from June 26, 2022 at 00:00:00 UTC and ended July 10, 2022 at 23:59:59 UTC. The AIS data included 150 class A vessels consisting of static information, dynamic information, and navigation information. The static information included the vessel identification numbers such as the Maritime Mobile Service Identity (MMSI) and International Maritime Organization (IMO) number, call sign and name, vessel types and dimensions, and the location of the electronic fixing device antenna. Static information are fixed and so rarely changes over time and are manually updated. The dynamic data included operational information related to navigation of the vessel. The data is collected at time intervals (data time stamp) and is automatically updated in accordance with navigational status of the vessel. The information (or data features), data types, and descriptions are given in Tables 1, 2, and 3 below.

Different AIS Messages

AIS Historical data for vessels contains data from two different types of AIS messages:

position messages and **static voyage messages**.

Message Types 1, 2, 3, 18, 19, 27 are some common position messages that contain values in the position related fields, such as speed, heading, rate of turn, position, and status of vessels. Details and descriptions are given in Table 3 below.

Message Types 5 & 24 are static voyage messages and contain values in the static details

and voyage related fields, such as identity, type, size and voyage information.

To identify the Name, IMO number or type of ships related to the AIS Position reports, the MMSI number must be used to join together the 2 different sets of AIS data.

For example, if position reports are recorded against MMSI 636018333 in AIS message type 1, then the IMO and name of that vessel is discovered by looking for AIS message type 5 reported using the same MMSI number as shown in Table 1 below:

Table 1: Spire AIS Message Type Reporting

Position					Both			Static				
speed	heading	rot	latitude	longitude	timestamp	msg_type	mmsi	imo	name	callsign	eta	destination
0	511	0	-86.6701	21.8061	2020-04-23T15:38:10.37	1	636018333					
					2020-04-23T15:58:40.38	5	636018333	9821299	SOUTHERN SHARK	D5PG4	2020-04-25T09:00:00	MX COA
13.5	288	-11	-86.7018	21.8523	2020-04-23T16:03:11.23	1	636018333					
12.8	290	0	-86.786	21.919	2020-04-23T17:30:50.34	27	636018333					
14.4	296	-128	-86.8096	21.92199	2020-04-23T17:38:00.41	1	636018333					
14.3	295	-43	-86.9663	21.9533	2020-04-23T17:48:51.11	1	636018333					
					2020-04-23T18:01:10.43	5	636018333	9821299	SOUTHERN SHARK	D5PG4	2020-04-25T12:00:00	MX COA
14.4	290	8	-86.9996	22.0199	2020-04-23T18:20:44.34	1	636018333					
14.1	300	0	-87.0682	22.0678	2020-04-23T18:28:00.45	1	636018333					

In the format of the acquired data, Spire has processed the AIS messages from each MMSI and joined together the most recently reported static values with each new position message for that MMSI providing a per vessel view of AIS historical data. A sample description is given in Table 2 below.

Table 2: ***AIS per Vessel Data Sample File Column Descriptions***

Column = Feature	Data Type	Description
created_at	*datetime*	ISO 8601 formatted timestamp in UTC of the time the vessel record was created
timestamp	String	ISO 8601 formatted timestamp in UTC of the time the AIS message was transmitted
static_updated_at	*datetime*	ISO 8601 formatted timestamp in UTC of the time the last AIS static message update was received
position_updated_at	*datetime*	ISO 8601 formatted timestamp in UTC of the time the last AIS position message update was received
mmsi	*integer*	The Maritime Mobile Service Identity of the vessel transmitting the AIS message *Possible values: 000000000 - 999999999*
latitude	float	*Vessel latitude in degrees (North = positive, South = negative) range -90 to +90*
longitude	float	Vessel longitude in degrees (East = positive, West = negative) range = -180 to +180
speed	*float*	Vessel speed over ground represented in knots *Possible values: 0 - 102.2 knots, 102.3 (not available)*
heading	*integer*	Vessel true heading in degrees *Possible values: 0 - 359 degrees, 511 (not available)*
course	*float*	Vessel course over ground in degree *Possible values: 0 - 359.9 degrees, 360.0 (not available)*
imo	*integer*	IMO number of the ship. A Unique International Maritime Organization number for the vessel that stays with the ship for its life. Valid values 7 digit number

name	*string*	Vessel name
call_sign	*string*	Vessel call sign
flag	*string*	Two-letter code for vessel flag
draught	*float*	Vessel draught represented in 1/10 meters *Possible values: 0.1 - 255, 0 (not available; default)*
ship_type_code	*integer*	Vessel ship and cargo type code. *Some common values: 30 (fishing vessel), 52 (tug boat), 70 (cargo/fishing ship)*
ship_type	*string*	De-coded vessel ship and cargo type
length	*integer*	Vessel length extracted from ship dimensions to_bow and to_stern in meters
width	*integer*	Vessel width extracted from ship dimensions to_port and to_starboard in meters
eta	*string*	Vessel estimated time of arrival as entered by the captain, represented in ISO 8601 format. *Possible values: Month: 1 - 12, 0 (not available; default); Day: 1- 31, 0 (not available; default); Hour: 0 - 23, 24 (not available; default); Minute: 0 - 59, 60(not available; default)*
destination	*string*	Vessel destination as entered by the vessel captain
status	*integer*	Vessel navigation status. *Some common values: 0 (under way using engine), 1 (at anchor), 3 (restricted maneuverability), 7 (engaged in fishing), 15*
collection_type	*string*	How the message was captured *Possible values: satellite or terrestrial or dynamic*

Table 3: Message types and descriptions

Message Number	Type of Report
1,2,3	Position Report
4	Base Station Report
5	Static and Voyage Related data
6	Binary address message
7	Binary acknowledgement
8	Binary broadcast message
9	Standard SAR aircraft position report
10	UTC/date inquiry
11	UTC/date response
12	Addressed safety related message
13	Safety related acknowledgement
14	Safety related broadcast message
15	Interrogation

16	Assignment mode command
17	DBNSS broadcast binary message
18	Standard Class B equipment position report
19	Extended Class B equipment position report
20	Data link management message
21	Aids-to-Navigation report
22	Channel management
23	Group assignment command
24	Static data report
25	Single slot binary message
26	Multiple slot binary message with Communications State
27	Position report for long range applications
x	Longitude
y	Latitude

Methodology Schemas

The statement of the problem for this project is thus, summarized as follows: given *n* spatial data points **X** in a *p*-dimensional metric space, partition the data points into say, *k* clusters such that the data points within a cluster are more similar to each other in proximity than data points in different clusters. That is, given a historical AIS dataset and/or real-time AIS dataset of maritime vessel activities, identify the Data Clustering Analytics Methods to model the tracking and monitoring of vessel geographical positions by clustering ships according to their proximities.

The approach for a solution to the problem is to apply Modified DBSCAN and Modified CURE algorithms to AIS data to monitor and track more accurately the movement and environment of maritime vessels on a global scale. Traditionally, the Euclidean distance formula has been paired with these algorithms and produces good results when examining a flat surface. Since the Earth is not flat, to improve the accuracy in measuring distance between vessels/individuals on Earth, we use the Haversine distance formula which evaluates navigating position distances using the great circle measurement. We also show how this will assist in the overall security for maritime monitoring and tracking.

The traditional DBSCAN clustering algorithm iterates from point to point to calculate the distances between points, identifies core points, and clusters the surrounding points together using the 2-D Euclidean Distance. Since the aim of our project is to cluster the AIS data collected globally so that the distance between vessels

across the hemispheres can be calculated, we implemented the three-dimensional Haversine distance metric. This is because the configuration of the global maritime network is organized along a circum-equatorial corridor-choke points- linking North America, Europe, and Pacific Asia through the Suez Canal, the Strait of Malacca, and the Panama Canal.

Likewise, the traditional CURE algorithm employs the 2-D Euclidean metric to find distances when applying the Clustering Using Representatives algorithm and for computing the minimum distance between representative points. Therefore, in this project, we have modified the traditional CURE algorithm to integrate the Haversine distance metric to handle the challenge of the even distribution of spatial datasets and the Pairwise distance matrix to speed the calculation of between clusters distances. Using the Haversine distance formula and the Pairwise distance matrix lends way to more accurate evaluation of distances globally. Therefore, the models are not confined to visualizing ships, objects, etc., in a grid section or smaller area such as a port or small inland waterway.

Throughout the project, the acquired Spire AIS dataset is of the form $X = \{X_1, X_2, X_3, \ldots, X_p\}$ where X_j ; j=1, 2, …, p are (n x 1) column vectors and X is an (n x p) matrix of the form $\{x_{i,j}\}$; i=1, 2, …, n and j=1, 2, …, p. The variables X_j; j = 1, 2, …, p are the features (vessel characteristics) and the rows of X are the objects (vessels) of the dataset. In particular, X is a dataset collection of information about maritime vessels navigating the large body of waters on earth such as rivers, lakes, seas, ocean, gulfs, lagoons, etc. The features are the characteristics of the vessels and the objects are the vessels describing the data point distributions in large body of waters around the world. Hence, for this project, we are concerned with the position locations of the vessels as they navigate the waters or in anchorage. Thus, the distances any particular vessel travels or the distances between vessels are of importance to this research. The orientation of the space of navigation is spherical and the space generated by the vectors X_j, j=1, 2, …, p is Euclidean. The objective is to employ machine learning analytics methods to identify the grouping of the vessels and tracking their trajectories as they navigate these large bodies of water in accordance with distance proximities and similarities in the features. Therefore, we will consider these groupings as a well-defined list or collection of objects with boundaries and therefore we shall treat them as sets. We shall be dealing with the cases of finite and countably infinite clusters. The collected AIS data downloaded and stored in file extensions .csv and .xlsx has the spreadsheet form and interpreted as:

Features → **Row #s ↓**	**cog**	**id**	**mmsi**	**…**	**x**	**y**
1	$X_{1,1}$	$X_{1,2}$	$X_{1,3}$	**…**	$X_{1,23}$	$X_{1,24}$
2	$X_{2,1}$	$X_{2,2}$	$X_{2,3}$	**…**	$X_{2,23}$	$X_{2,24}$
.	.	.	.	**…**	.	.

.	.	.	.	…	.	.
.	.	.	.	…	.	.
n	$X_{n,1}$	$X_{n,2}$	$X_{n,3}$	…	$X_{n,23}$	$X_{n,24}$

This we represent in matrix notation as:

$$X = \begin{bmatrix} x_{1,1} & x_{1,2} \cdots \cdots & \cdots x_{1,24} \\ x_{2,1} & x_{2,2} \cdots \cdots & \cdots x_{2,24} \\ \vdots & \vdots \quad \cdots \cdots & \cdots \quad \vdots \\ x_{i,1} & x_{i,2} \ldots x_{i,j} & \ldots x_{i,24} \\ \vdots & \vdots \quad \cdots \quad \cdots & \cdots \quad \vdots \\ x_{n,1} & x_{n,2} \ldots x_{n,j} & \ldots x_{n,24} \end{bmatrix}$$

Machine Learning Process

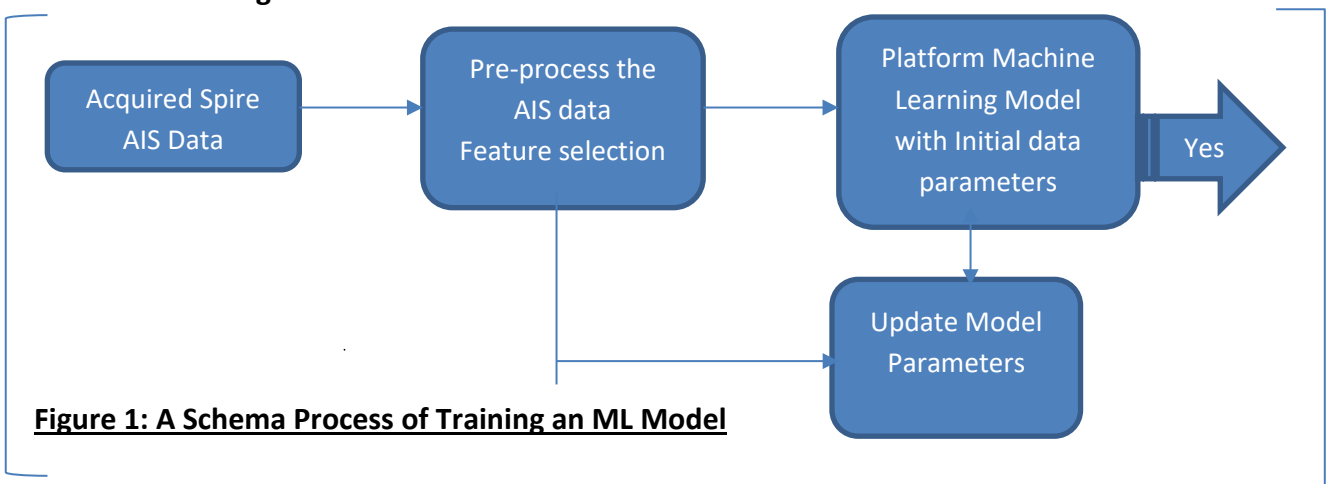


Figure 1: A Schema Process of Training an ML Model

AutoML Process

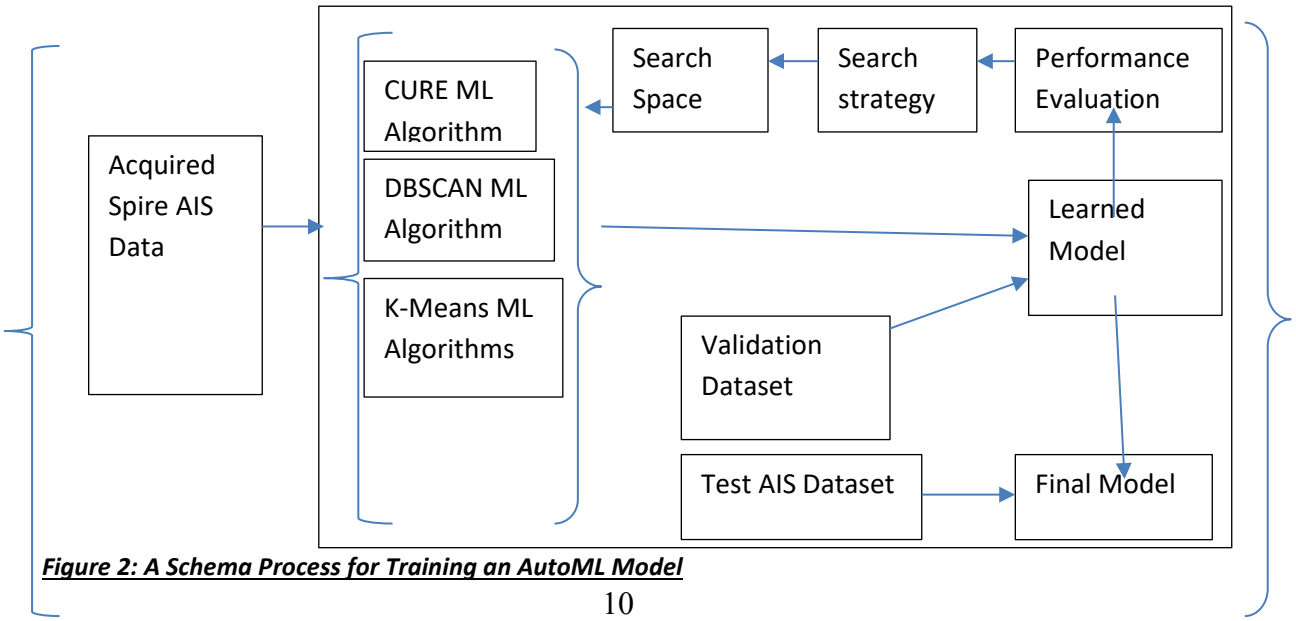


Figure 2: A Schema Process for Training an AutoML Model

Pairwise Distance Matrix Calculations

We denote the pairwise distance function as *PdistHaversine* (XI, XJ) and its output as D where XI and XJ are (m × 2) dataset of latitude/longitude points pairings of vessel positions. The first columns of XI and XJ are latitudes (in degrees) and the second columns are longitudes (in degrees) of a vessel position as collected by the AIS units and decoded for computational analysis. Then D←PdistHaversine (XI, XJ) results in the (m × m) dissimilarity distance matrix D. The distance is measured in kilometers and the Earth is approximated as a sphere. The Haversine formula is used for evaluation, thus the method is accurate on small distances but inaccurate when points are diametrically on the opposite sides of the sphere. Thus, the Haversine distance metric is more accurate for tracking vessels in open sea and ocean waters and at anchoring stations with local data. The pairwise distance matrix allows the algorithm to calculate the distances between vessels by traversing the matrix rather than by iterating from point to point to calculate the distances between points. This reduces computational time.

Table 4: Haversine Distance Calculation

lat1 = XI(:,1)*pi/180
lon1 = XI(:,2)*pi/180
lat2 = XJ(:,1)*pi/180
lon2 = XJ(:,2)*pi/180
% corresponding respectively to the latitude/longitude of vessels one and two
% We next create a coordinate system of latitude and longitude as follows:
[LAT1, LAT2] < --- > meshgrid (lat1, lat2)
[LON1, LON2] < --- > meshgrid (lon1, lon2)
% Subsequently, in this new coordinate system, we let:
$\Delta\theta = LAT2 - LAT1$
$\Delta\varphi = LON2 - LON1$
Thenceforth, the Haversine distance metric in the new coordinates is given by:
A = sin (DLAT/2) ^2 + cos(LAT1) *cos(LAT2) *(sin(DLON/2) ^2)
C = 2 * atan2 (sqrt(A), sqrt(1-A))
D = R * C,
% where D is the distance between any two vessels

Then, given a data matrix X of vessel positions in latitude/longitude measured in degrees, we have that D = *PdistHaversine*(X) gives the (n x n) squared dissimilarity matrix of distances between vessels.

A modified DBSCAN and CURE Algorithms

The modified pseudocode of the DBSCAN algorithm and the implementation schema of the CURE algorithm are given in Table 5 and Figure 3 below, respectively.

Table 5: Modified Density-Based Spatial Clustering Algorithm

```
Read all lat lon points
Plot all points
# Crop the data to contain the points within the bounding box
selectedVessel ← based on lat and lon positions
LATbound, LONbound ← r
Bounding_Box = selectedVessel ± r
included_vessel_points ← all points within Bounding Box
plot (included_vessel_points)
# Density-based clustering
D ← sklearn.metrics.pairwise.distances(included_vessel_points)
index, CorePts = sklearn.cluster.DBSCAN (D, epsilon, MinPts, 'Distance')
plot (included_vessel_points, color=index)
# Identify number of outliers and number of core points
total_outliers = sum (index == -1)
total_core_points = sum (CorePts == 1)
```

DBSCAN was performed using a matrix of pairwise distances between observations as input of a defined DBSCAN function in the library of the platform of choice and found the number of outliers and core points for tracked vessels. Prior to implementing the program, we used the k-distance graph approach to determine the suitable values for the epsilon neighborhood. The following pseudo code was implemented in MATLAB to determine the appropriate epsilon value:

Table 6: K-Distance Graph Approach to Determine Epsilon Value

```
K-Distance Graph Approach
# Find the minpts smallest pairwise distances in ascending order
kD ← pdistHaversine(appropriate arguments)
# Plot the k-distance graph by calling the MATLAB® plot and sort    #built in functions
plot (sort (kD(All_points)))
```

Modified CURE Implementation

Clustering Using Representatives (CURE) is a hierarchical based clustering algorithm. The steps involved in the modified algorithm are given in Figure 3 below.

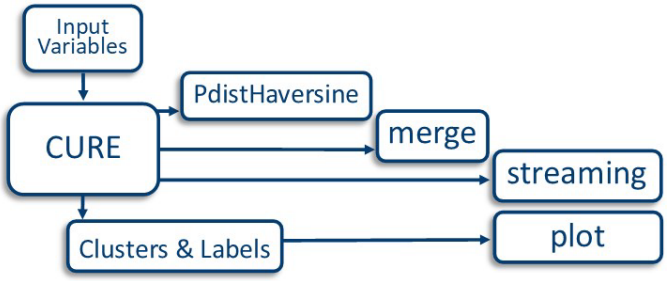


Figure 3: Modified CURE workflow overview

The Modified CURE function is a multivariate function with four input values and we denote it as (CURE (X, α, c, k)), where **X** is the AIS data set, α is the shrinking factor, c is the number of representative points, and k is the desired number of clusters. The function's outputs are the labels and clusters. The function calls auxiliary functions struc('point','rep','mean','index'), the PdistHaversine, the merge function, the streaming function, and the plot function. The user interface code first loads the data file and the parameters (α, c, and k), calls CURE, then the plot function for the outcome.
Modified CURE clustering algorithm was performed using a matrix of pairwise distances between observations as input of a defined CURE function in the platform library to find the clusters, number of outliers and core points of tracked vessels. Using the same downloaded data as with the DBSCAN algorithm, the latitude/longitude position point pairings were extracted and stored in .csv and .xlsx file extensions and in ASCII format. The data was extracted without any particular attention to geographical regions or the body of waters. Since our intention here is to create a program that is capable of tracking and monitoring maritime vessels and their surroundings based on using the correct distance formula for a spherical surface applied to clustering algorithms globally.

3. Results/Findings

DBSCAN Algorithm Implementation

The DBSCAN program was implemented for a sample of the downloaded AIS data with n = 31323 using a Dell intel CORE i7 to take advantage of the multicore processor parallel computing capability of MATLAB R2023b. The outcome for epsilon values of 1, 1.55 and 2 set for the minpts of 50 are given in Figure 4 below.

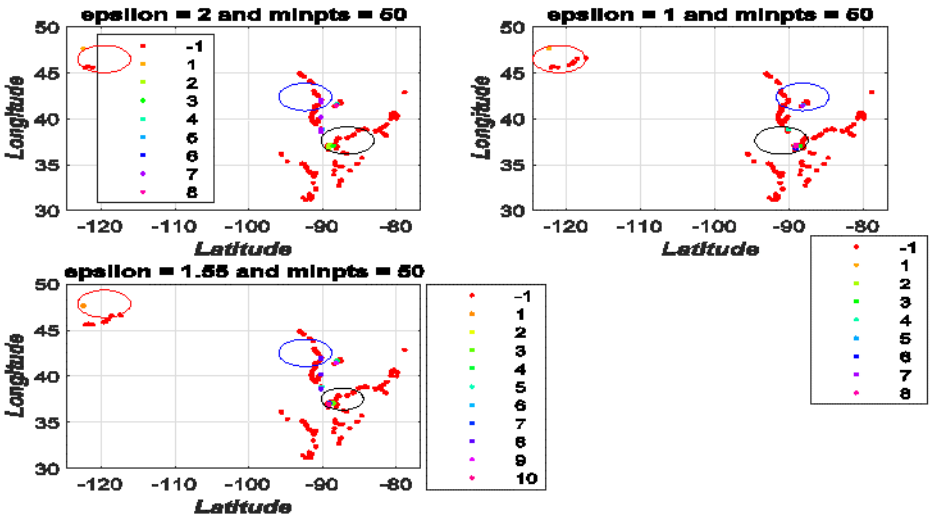


Figure 4: DBSCAN Identifies sets of clusters and sets of outliers/noise points and tracking objects ellipsed.

DBSCAN Findings

As can be discerned from Figure 4 above, when the epsilon value is set at the maximum allowed by the K-nearest graph of 2 and a minimum number of core points (minpts) of 50, DBSCAN under our model was able to identify eight distinct clusters such as those highlighted by the red, black and blue ellipses which show densities of vessels in close proximity and 1942 outliers or objects with DBSCAN index of -1 and 124 core points or objects with DBSCAN index of 1. When the epsilon value was reduced to 1, DBSCAN still identified eight distinct clusters with the same number of outliers and core points. However, when the epsilon value was set at 1.55, DBSCAN identified 10 distinct clusters and 1988 outliers or objects with DBSCAN index of -1 and 131 core points.

The minpts were set at 5, 10, 15, 20, 50, and 100. The minpts at 5 produced 195 clusters for epsilon = 1, 164 clusters for epsilon= 1.55 and 152 clusters for epsilon = 1.55 with a total sum of 2,869 core points. The DBSCAN index = 1 and each epsilon neighborhood was 6. At the minpts of 10, DBSCAN produced an average of 80 clusters for each epsilon setting of 1, 1.55, and 2 with a total of 2,345 core points. The number of clusters continue to decrease with increasing value of minpts and at the minpts = 100, there were one, three, and five clusters for epsilon settings of 1, 1.55, and 2 respectively with a total of 378 core points.

Thus, even though the number of minimum points must be greater than the dimension of the input data matrix plus one and that larger values of minpts are usually recommended for data set with noise, we see from these results that the minpts of 50 have yielded a more significant number of clusters. More significant clusters will assist in

targeting areas associated with an individual vessel because the more clusters found the closer those clusters will be to containing the number of vessels input as the minpts. The algorithm developed for this project has the advantage of implementation over the global AIS database of regional databases or local focus on shore anchorages or harbors.

The CURE Algorithm Implementation

The modified CURE clustering algorithm was performed using a matrix of pairwise distances between observations as input of a defined MATLAB CURE function to find the clusters, number of outliers and core points of tracked vessels. Using the same downloaded data as with the DBSCAN algorithm, the latitude/longitude position point pairings were extracted and stored in .csv and .xlsx file extensions format. The data was extracted without any particular attention to geographical regions or the body of waters. Since our intention here is to create a program that is capable of tracking and monitoring maritime vessels and their surroundings based on using the correct distance formula for a spherical surface applied to clustering algorithms.

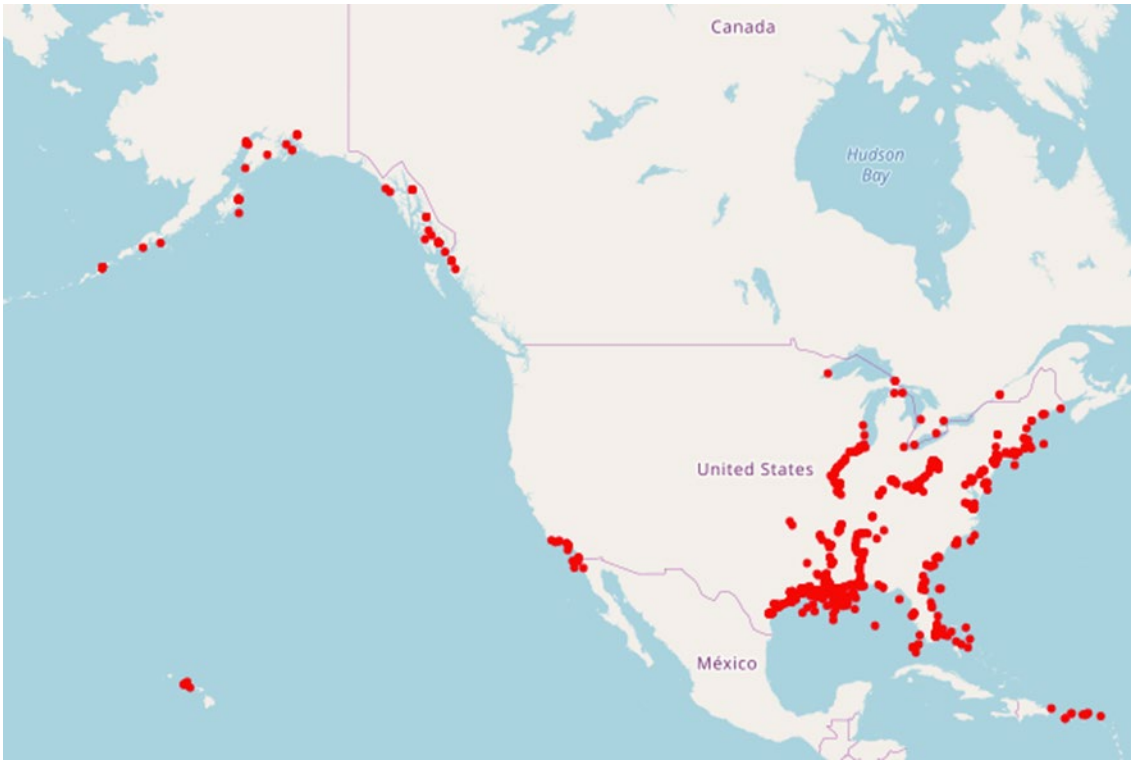


Figure 5: A Cluster of 1000 of the 21,694 vessels of the processed AIS SPire data

Figure 6: Clusters of maritime vessels tracked inland

Figure 7: Clusters of maritime vessels tracked in open seas

CURE Findings:

As can be seen in, Figure 5, Figure 6, and Figure 7 above, the data used for this project in implementing the cure algorithm is primarily in the eastern part of the United States of America. It includes inland waterways and open sea tracked vessels. For the CURE algorithm, the latitude and longitude pairs database were set to **X**. Through trial and error, we set the shrinking factor, alpha (α), to 0.51, the number of representative points, c, to 9, and the desired number of clusters to 8. Therefore, there were nine (9) representative points assigned to outline each cluster which were generated from shrinking initial cluster longitude and latitude point pairs by 0.51 percent. Eight (8) clusters formed as seen in Figure 5 and Figure 6. Smaller or clusters more focused on a specific area can be formed by increasing k and including longitude/latitude point pairs for **X** that are extracted from the database by concentrating on a specific area. The algorithm introduced in this project has the advantage of application throughout the global AIS database of vessels tracked inland and in the open seas but can also be localized to zoom in on a specific location or region.

The XGBoost, EDA, and FASTAI Models- Tracking and Building Shipping Routes

In order to equipped our AutoML model with as many as possible clustering methods, we also implemented the XGBoost , EDA and FASTAI Algorithms on the acquired Spire AIS data.

XGBoost (Extreme Gradient Boosting) is a popular and powerful machine learning algorithm known for its high performance in various data science and machine learning tasks. It belongs to the family of gradient boosting algorithms and has gained widespread adoption due to its effectiveness in handling structured/tabular data and feature engineering. Due to its versatility and effectiveness, XGBoost is commonly used in various machine learning tasks, such as regression, classification, ranking, and recommendation systems. It has also found applications in domains such as finance, healthcare, marketing, and many others where structured data analysis is essential.

EDA + FASTAI
Exploratory Data Analysis (EDA) is the process of visually and statistically exploring and summarizing a dataset to gain insights into its structure, relationships, and characteristics. EDA is typically performed at the beginning of a data analysis project to understand the data, identify patterns, detect outliers, and prepare for subsequent modeling tasks.

Fastai is an open-source deep learning library built on top of PyTorch. It aims to make deep learning more accessible to practitioners by providing high-level abstractions and best practices. Fastai provides a simple and intuitive API for creating and training deep learning models with minimal code.

While EDA and fastai are distinct concepts, they can be used together in a typical data science or machine learning workflow. The EDA process helps data scientists understand the dataset they are working with and make decisions about data preprocessing and feature engineering. Once the data is ready, fastai can be utilized to build and train deep learning models for various tasks like image classification, natural language processing, and more.

For this project we have used the acquired Spire Maritime AIS Data for shipping route data.
In this case, we have adopted a two-step approach in which we looked at XGBoost, EDA, and FASTAI separately and see how they can complement each other to develop a more efficient shipping routing, speed, loading, and deployment of vessels. The two-prong will involve:

- Port data collection and analysis.
- Route data collection and analysis.

Port Data Analysis

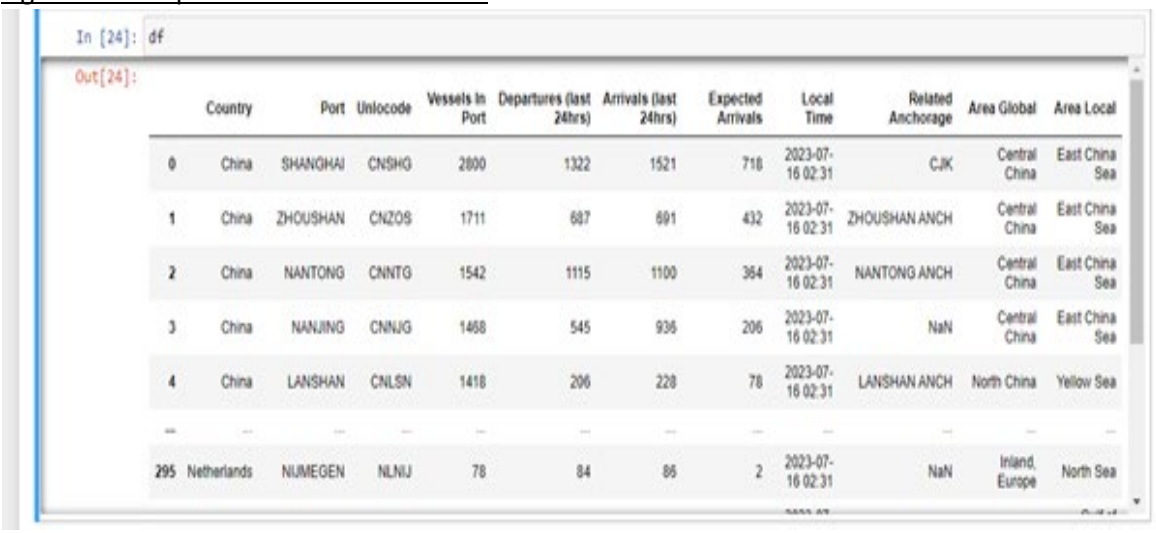
This report uses shipping data from ports around the world to analyze and develop models for feature importance. This process involves:

- Preprocessing
- Detect continuous and categorical variables.
- Normalize and impute data.
- For every target variable in the dataset:
- Compare performance on 27 models + a TabNet model.
- Output model performance and processed data in CSV format for every dataset.
- Save plots + CSVs of XGBoost Feature Importances.
- Save best performing FastAI model.

In this study, we undertake EDA and FastAI preprocessing, model comparison, make predictions, extra feature importance and look for leaky features.

In this part of the project, we decided to use the port data for July 16, 2023, extracted from The World Bank Data Catalog on global international ports for localized and choke points arrival and departures.

Figure 8: A snapshot of Port Data Format



```
In [24]: df
```

Out[24]:

	Country	Port	Unlocode	Vessels In Port	Departures (last 24hrs)	Arrivals (last 24hrs)	Expected Arrivals	Local Time	Related Anchorage	Area Global	Area Local
0	China	SHANGHAI	CNSHG	2800	1322	1521	718	2023-07-16 02:31	CJK	Central China	East China Sea
1	China	ZHOUSHAN	CNZOS	1711	687	691	432	2023-07-16 02:31	ZHOUSHAN ANCH	Central China	East China Sea
2	China	NANTONG	CNNTG	1542	1115	1100	364	2023-07-16 02:31	NANTONG ANCH	Central China	East China Sea
3	China	NANJING	CNNJG	1468	545	936	206	2023-07-16 02:31	NaN	Central China	East China Sea
4	China	LANSHAN	CNLSN	1418	206	228	78	2023-07-16 02:31	LANSHAN ANCH	North China	Yellow Sea
...	...	...	...	...	...	...	...	...	...	...	...
295	Netherlands	NIJMEGEN	NLNIJ	78	84	86	2	2023-07-16 02:31	NaN	Inland Europe	North Sea

Figure 9: A snapshot of Port Data Format

	Unnamed: 0	Vessels in Port	Departures(Last 24 Hours)	Arrivals(Last 24 Hours)	Expected Arrivals
count	480.000000	480.000000	480.000000	480.000000	480.000000
mean	239.500000	153.312500	98.981250	108.662500	39.233333
std	138.708327	217.297037	170.504574	185.357564	82.385289
min	0.000000	51.000000	0.000000	1.000000	0.000000
25%	119.750000	63.000000	26.750000	30.000000	3.000000
50%	239.500000	86.000000	48.000000	56.000000	16.000000
75%	359.250000	144.000000	102.000000	106.250000	40.250000

Figure 10: A Snapshot of Summary report of port data

Figure 11: A Snapshot of Countries involve in port study

Figure 12: A Snapshot of recorded vessel in port

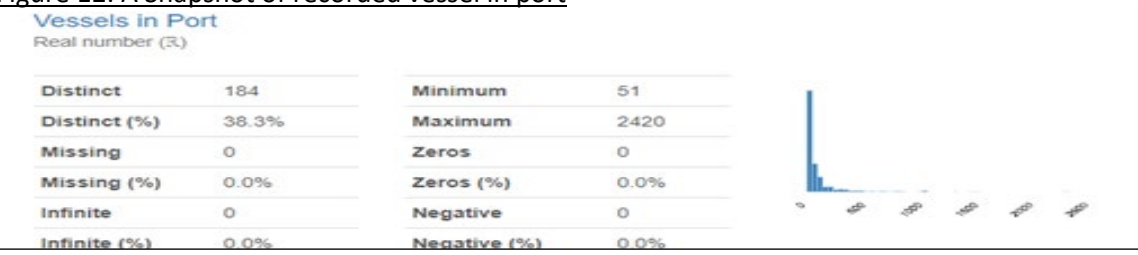


Figure 13: Interactions expected arrivals and departures in 24 hours

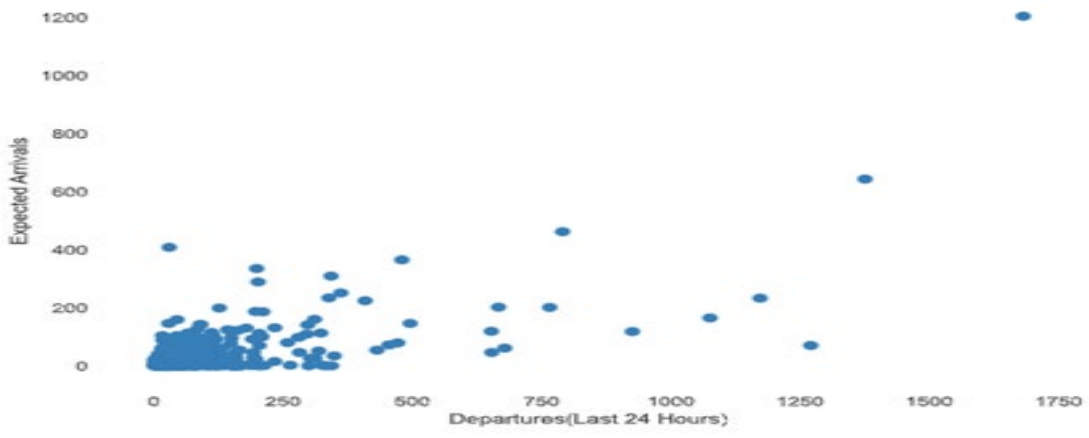
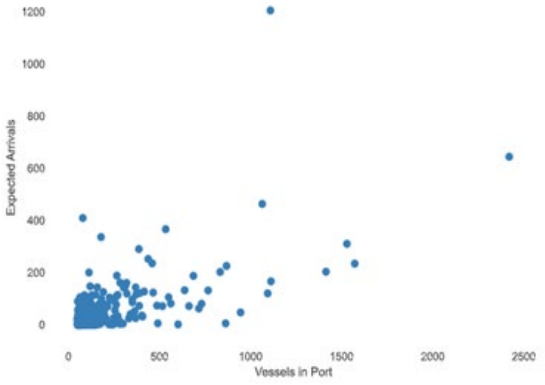
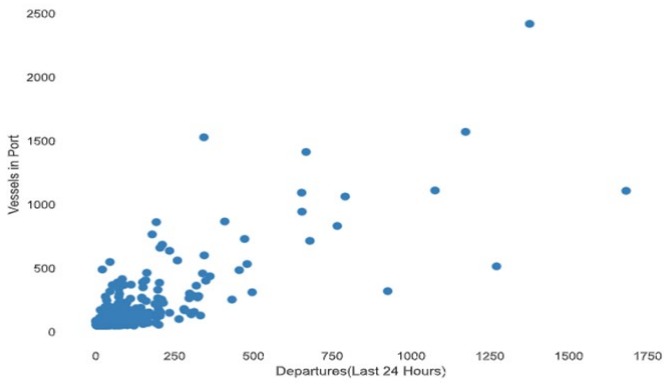
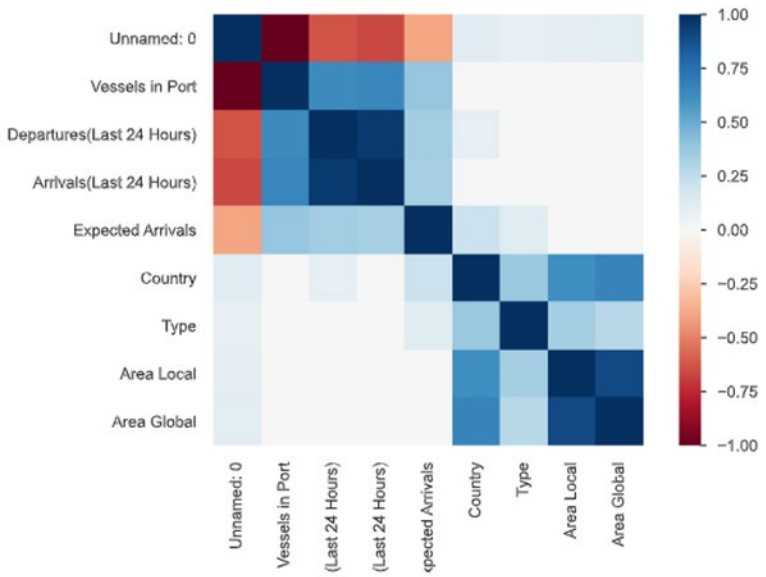


Figure 13: Interactions expected arrivals and Departures in 24 hours



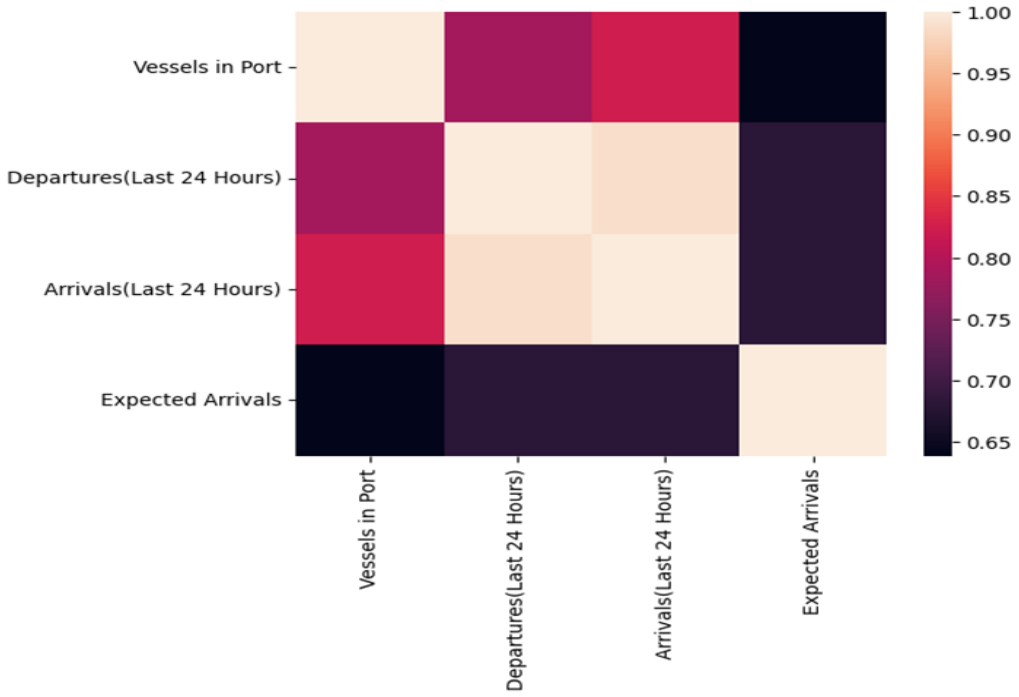


Correlation Data



Unnamed: 0	Vessels in Port	Departures(Last 24 Hours)	Arrivals(Last 24 Hours)	Expected Arrivals	Country	Type	Area Local	Area Global	
Unnamed: 0	1	-1	-0.63	-0.657	-0.398	0.113	0.079	0.088	0.106
Vessels in Port	-1	1	0.628	0.656	0.39	0	0	0	0
Departures(Last 24 H	-0.63	0.628	1	0.959	0.349	0.082	0	0	0
Arrivals(Last 24 Hou	-0.657	0.656	0.959	1	0.33	0	0	0	0
Expected Arrivals	-0.398	0.39	0.349	0.33	1	0.216	0.124	0	0
Country	0.113	0	0.082	0	0.216	1	0.374	0.616	0.663
Type	0.079	0	0	0	0.124	0.374	1	0.34	0.279
Area Local	0.088	0	0	0	0	0.616	0.34	1	0.902

	Vessels in Port	Departures(Last 24 Hours)	Arrivals(Last 24 Hours)	Expected Arrivals
Vessels in Port	1.000000	0.786678	0.823399	0.638791
Departures(Last 24 Hours)	0.786678	1.000000	0.988624	0.679128
Arrivals(Last 24 Hours)	0.823399	0.988624	1.000000	0.679451
Expected Arrivals	0.638791	0.679128	0.679451	1.000000



XGBoost Findings:

XGBoost Predictions

Target Variable: Expected Arrivals

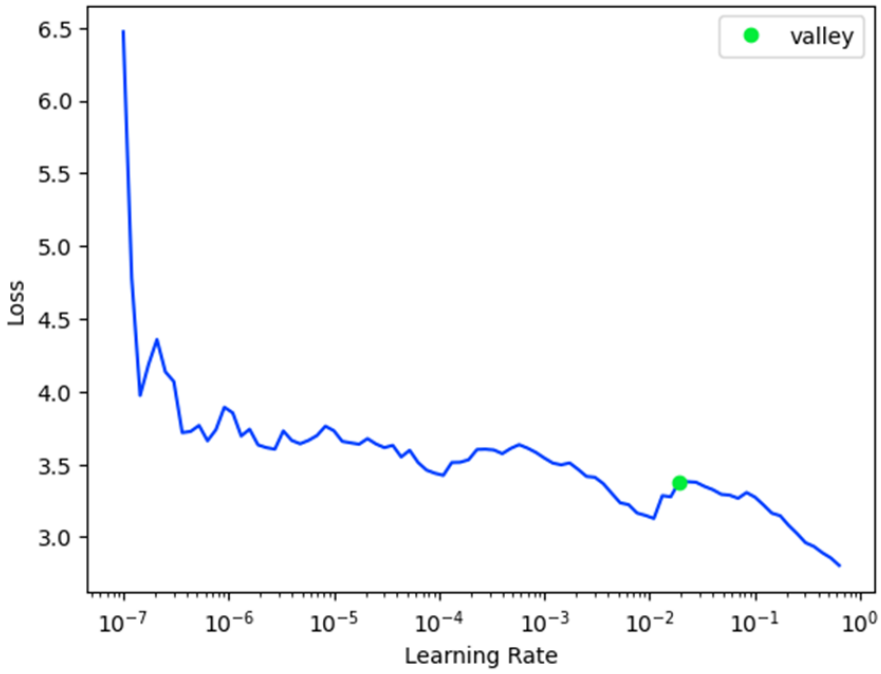
LEARNING RATE: 0.1

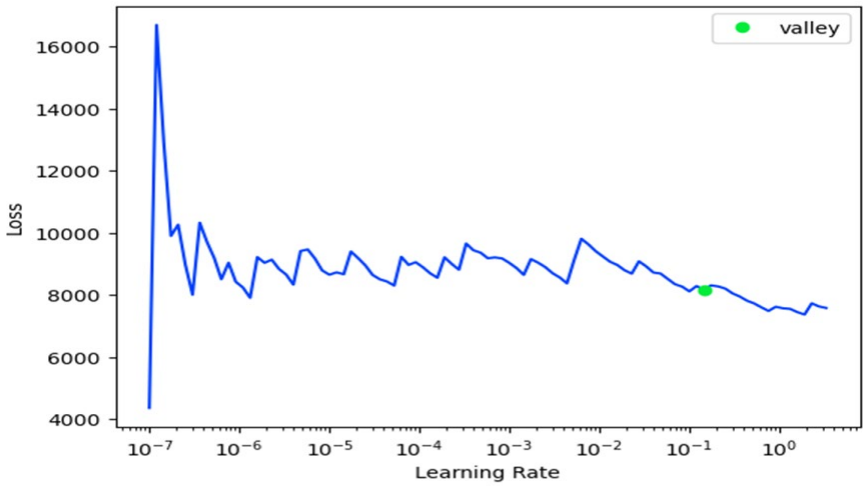
epoch	train_loss	valid_loss	_rmse	time
0	8664.298828	6937.503906	83.291679	00:00
1	8111.142090	5894.808105	76.777657	00:00
2	7305.997559	15502.333008	124.508362	00:00

```
Better model found at epoch 0 with _rmse value: 83.29167938232422.
Better model found at epoch 1 with _rmse value: 76.77765655517578.
No improvement since epoch 1: early stopping
```

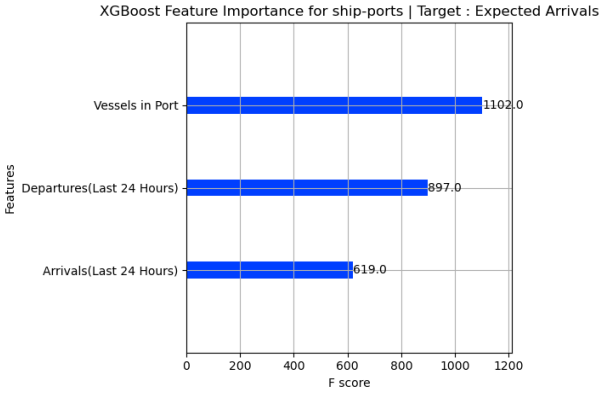
	Type	Area Local	Area Global	Vessels in Port	Departures (Last 24 Hours)	Arrivals (Last 24 Hours)	Expected Arrivals	Expected Arrivals_pred
0	1.0	1.0	1.0	-0.373504	-0.400292	-0.413809	10.0	4.348602
1	1.0	1.0	1.0	-0.306378	-0.411740	-0.403239	19.0	4.067498

	Type	Area Local	Area Global	Vessels in Port	Departures (Last 24 Hours)	Arrivals (Last 24 Hours)	Expected Arrivals	Expected Arrivals_pred
2	1.0	1.0	1.0	-0.324278	-0.245744	-0.181263	0.0	3.852627
3	1.0	1.0	1.0	1.770047	-0.325880	-0.149553	105.0	16.032272
4	1.0	1.0	1.0	-0.382454	-0.423188	-0.440235	37.0	4.470521
5	1.0	1.0	1.0	-0.266103	-0.348776	-0.350387	36.0	3.784293
6	1.0	1.0	1.0	-0.445105	-0.491876	-0.508941	2.0	5.022128
7	1.0	1.0	1.0	-0.355604	-0.457532	-0.493086	0.0	4.444355
8	1.0	1.0	1.0	-0.163176	0.275139	0.188696	5.0	5.878642






```
XGBoost Predictions vs Actual==========
   actual  predicted
0      22  45.332806
1      20  37.001637
2      16  36.442001
3       1  12.142123
4      20  18.585485
XGBoost RMSE:  72.624405
```



Target Variable: Departures(Last 24 Hours)

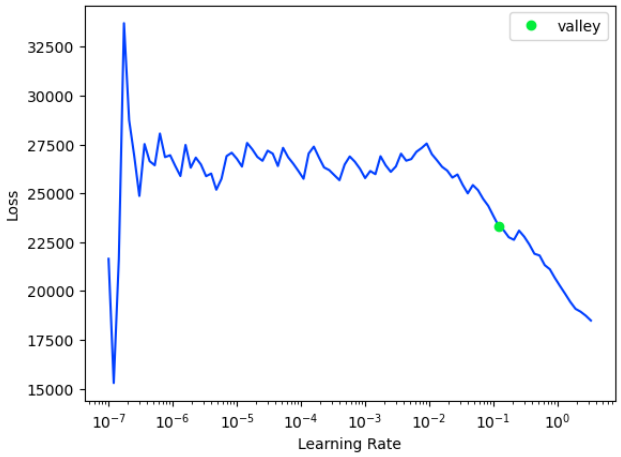
```
LEARNING RATE: 0.1
```

epoch	train_loss	valid_loss	_rmse	time
0	26091.853516	67996.257812	260.760925	00:00
1	24606.947266	45143.000000	212.468826	00:00
2	21573.218750	9154.762695	95.680527	00:00
3	17498.271484	1244061.375000	1115.375000	00:00

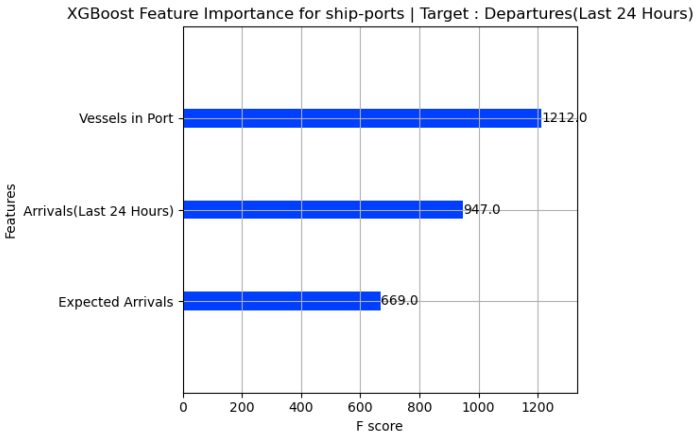
```
Better model found at epoch 0 with _rmse value: 260.76092529296875.
Better model found at epoch 1 with _rmse value: 212.4688262939453.
Better model found at epoch 2 with _rmse value: 95.68052673339844.
No improvement since epoch 2: early stopping
```

	Type	Area Local	Area Global	Vessels in Port	Arrivals(Last 24 Hours)	Expected Arrivals	Departures(Last 24 Hours)	Departures(Last 24 Hours)_pred
0	1.0	1.0	1.0	-0.313052	-0.501506	-0.394961	29.0	22.413132
1	1.0	1.0	1.0	-0.474150	-0.585816	-0.260466	8.0	18.139421
2	1.0	1.0	1.0	0.428002	-0.287487	-0.200691	35.0	38.745789
3	1.0	1.0	1.0	-0.409711	-0.118866	-0.499568	84.0	42.674797
4	1.0	1.0	1.0	-0.165761	-0.073468	-0.350129	60.0	44.126987
5	1.0	1.0	1.0	-0.409711	-0.255060	0.292455	60.0	55.226303
6	1.0	1.0	1.0	-0.349874	0.140551	0.412006	117.0	75.422394

	Type	Area Local	Area Global	Vessels in Port	Arrivals(Last 24 Hours)	Expected Arrivals	Departures(Last 24 Hours)	Departures(Last 24 Hours)_pred
7	1.0	1.0	1.0	6.526738	7.696071	2.922570	1173.0	1750.823853
8	1.0	1.0	1.0	-0.386697	-0.358826	-0.544399	38.0	21.840189




```
XGBoost Predictions vs Actual==========
   actual   predicted
0      79   68.106300
1      29   15.541866
2      31   45.384262
3      84   177.082687
4      48   36.029919
XGBoost RMSE:  76.02424
```

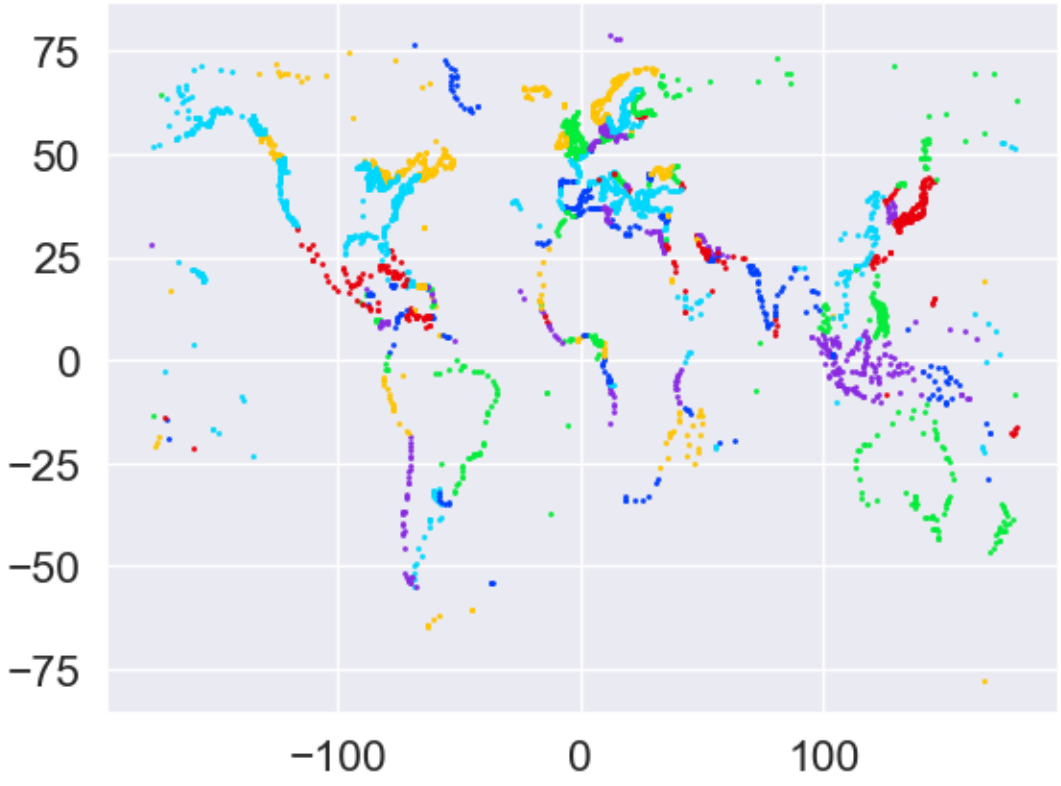


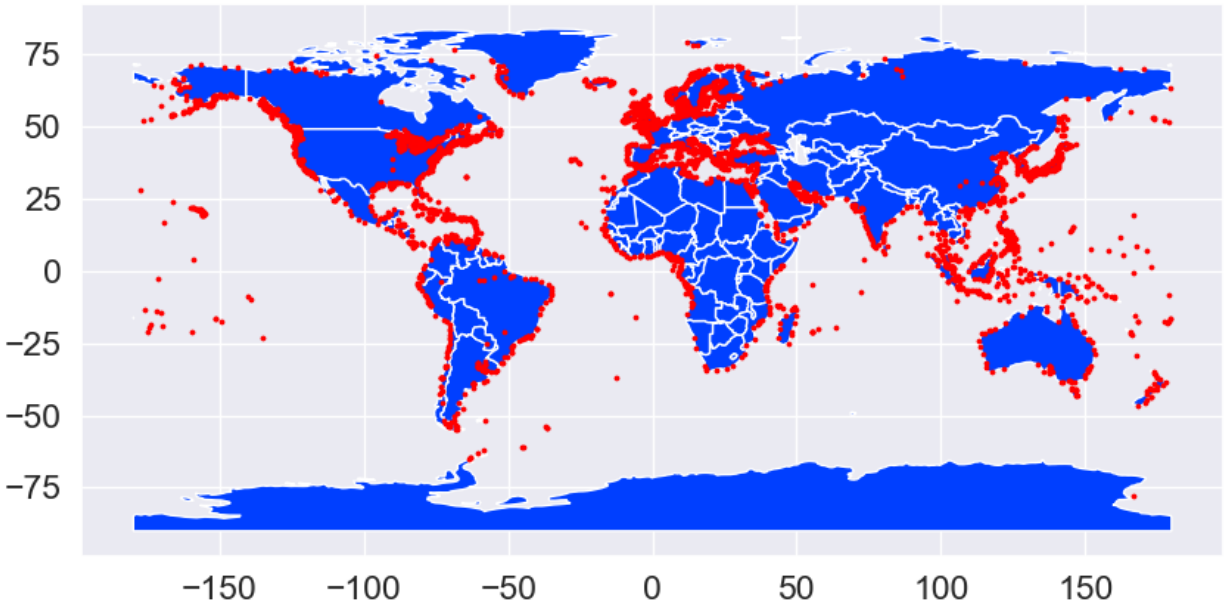
Route Determination

Shipping route determination involves the process of planning and selecting the optimal routes for maritime vessels to transport goods from one location to another. This process is crucial for efficient and cost-effective maritime transportation. Factors such as distance, weather conditions, navigational constraints, cargo type, port availability, and safety considerations all play a significant role in determining shipping routes. Since these factors will cause some restrictions in one way or the other, it is important to look at routes from various angles. Therefore, in this project, we have looked at the shipping routes based solely off distance and efficiency, and we also looked at all possible routes from shipping ports in an area.

Distance and Efficiency Routing

According to our calculation, there are over 3579 ports around the world demonstrated in the map we created in python below.





The distance and path through sea mass from port to port can be determined through various python packages. For this project, we use scgraph python library function written by Connor Makowski and the MIT Supply Chain CAVE lab team to derive possible routes and the distance of the developed route(s).

Example of routes:

Length: 21322.8587 km

Length: 21396.0937 km

Length: 18089.8034Km

All Possible Routes Calculation

Another calculation which was done is deriving and calculating all possible routes in an area or from a set of points. This calculation is useful to determine efficiencies in ships that are executing multiply location stops throughout their journey. This stopping of ships can be triggered by weather avoidance, refueling, maintenance/repairs, and safety. This was done by the creation of an un-directed weighted graph using all the points/coordinates connected to each other. This graph is then sorted with "Dijkstra's Algorithm" to find the shortest path to all of the points.

This is an example of 12 randomly selected ports around the world which is placed in a list in python. Now all possible routes will be determined, and the shortest path will be outlined and the distance of the path.

```
[[39.316667, 26.7, 1],
 [12.25, 109.233333, 2],
 [21.95, -159.35, 3],
 [22.0, -159.333333, 4],
 [-13.283333, -176.133333, 5],
 [36.9, 21.666667, 6],
 [45.216667, 14.55, 7],
 [36.766667, 3.066667, 8],
 [35.133333, -120.65, 9],
 [25.15, 52.866667, 10],
 [57.466667, -135.05, 11],
 [58.233333, -134.266667, 12]]
```

Results of All possible path

All points are labeled and each path is given a weight based on the length as shown below.

```
-------------------------------------
Path 1 = [1, 6, 7, 8, 10, 2, 9, 12, 11, 4, 3, 5]
Weight = 1214.9622863877134
-------------------------------------
Path 2 = [2, 10, 1, 6, 7, 8, 9, 12, 11, 4, 3, 5]
Weight = 1996.1360876162607
-------------------------------------
Path 3 = [3, 4, 5, 9, 11, 12, 8, 7, 6, 1, 10, 2]
Weight = 1355.1382599229205
-------------------------------------
Path 4 = [4, 3, 5, 9, 11, 12, 8, 7, 6, 1, 10, 2]
Weight = 1354.9389191309683
-------------------------------------
Path 5 = [5, 3, 4, 9, 11, 12, 8, 7, 6, 1, 10, 2]
Weight = 1634.2589069252747
-------------------------------------
Path 6 = [6, 1, 7, 8, 10, 2, 9, 12, 11, 4, 3, 5]
Weight = 1184.5249276175823
-------------------------------------
Path 7 = [7, 6, 1, 8, 10, 2, 9, 12, 11, 4, 3, 5]
Weight = 1167.61373707686
-------------------------------------
Path 8 = [8, 7, 6, 1, 10, 2, 9, 12, 11, 4, 3, 5]
Weight = 1131.332729539668
-------------------------------------
Path 9 = [9, 11, 12, 4, 3, 5, 8, 7, 6, 1, 10, 2]
Weight = 1162.624880032972
-------------------------------------
Path 10 = [10, 1, 6, 7, 8, 2, 9, 12, 11, 4, 3, 5]
Weight = 1426.4898010371962
-------------------------------------
Path 11 = [11, 12, 9, 4, 3, 5, 8, 7, 6, 1, 10, 2]
```

```
Weight = 1245.193446876691
-------------------------------------
Path 12 = [12, 11, 9, 4, 3, 5, 8, 7, 6, 1, 10, 2]
Weight = 1244.6344022836881
-------------------------------------
The shortest path to all nodes is: [8, 7, 6, 1, 10, 2, 9, 12, 11, 4,
3, 5]
The weight of the path is: 1131.332729539668
```

Tracking Specific Vessels

The advantage of knowing which vessels are in a specified area or should be in a specified area lends its way to controlling the safety of people travelling on marine vessels, the transportation of cargo or in carrying out critical military missions. The ability to detect unauthorized vessels or objects in waterways will always be a greatly appreciated and very useful tool. This is the reason for further research to identify clusters of vessels using the correct distance measurement to improve previous tracking techniques.

Geoscatter plot and Folium were the two visualization tools used to identify maritime vessels. Geoscatter plot with Basemaps is a tool in MATLAB for plotting on world map.

Geoscatter plots latitude, longitude and can also be programmed to include labels (text). The following maps were created using the Geoscatter plot tool.

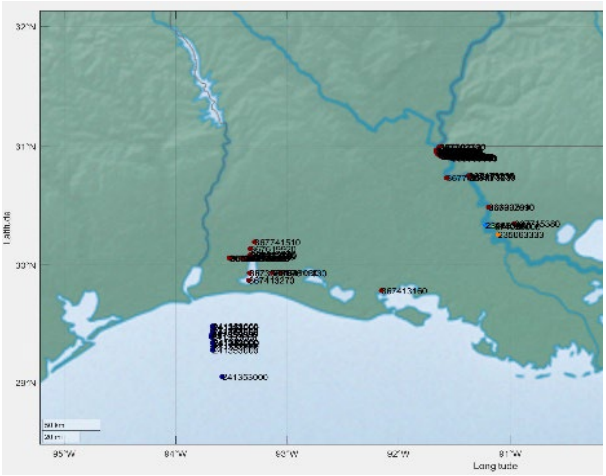


Figure 14: MATLAB Geoscatter plot zoomed to the southwest coast of the USA. It includes MMSIs identifying vessels.

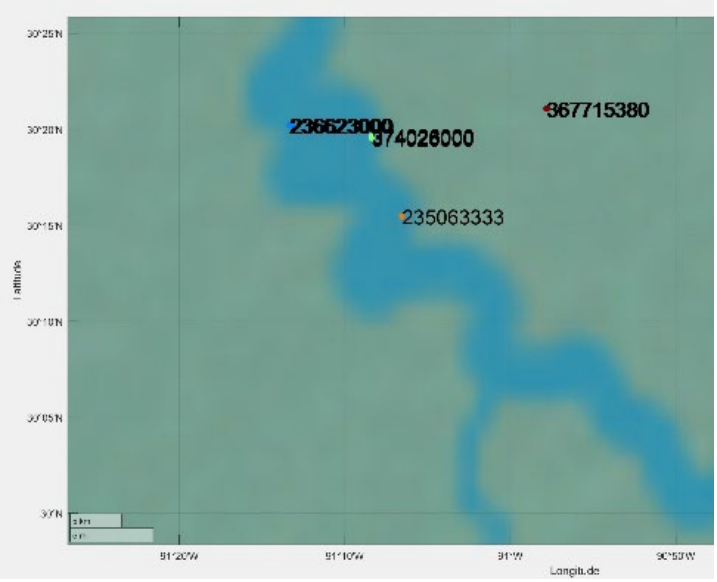


Figure 15: MATLAB Geoscatter plot zoomed to the Mississippi River, including MMSIs to identify vessels.

Figure 14 and Figure 15 plots show clusters, after applying the modified DBSCAN or CURE clustering algorithms, with the marine vessels' identification number, MMSI, which is unique to each vessel. Due to the distortion of the plots produced in MATLAB, the plots were also produced in Python. An example can be viewed in Figure 16 below.

Folium is a library function in Python that is used to access the open-source JavaScript library, Leaflet. Parameters that can be indicated when coding Folium include location (center point coordinate for requested map), width and height of map, minimum and maximum zoom, zoom_start (denotes how far in or out you want the zoom to be set at initially, etc. Folium also allows popups for markers on the map which consists of information desired to be shown when the marker is clicked. We have used these makers to indicate latitude, longitude and the MMSI (identification number) of vessels tracked. Figure 17 and Figure 18 plots were created using the Folium library function. These plots just show the first two vessels of the subset of the AIS data after applying the Modified DBSCAN and Modified CURE algorithms to cluster the data.

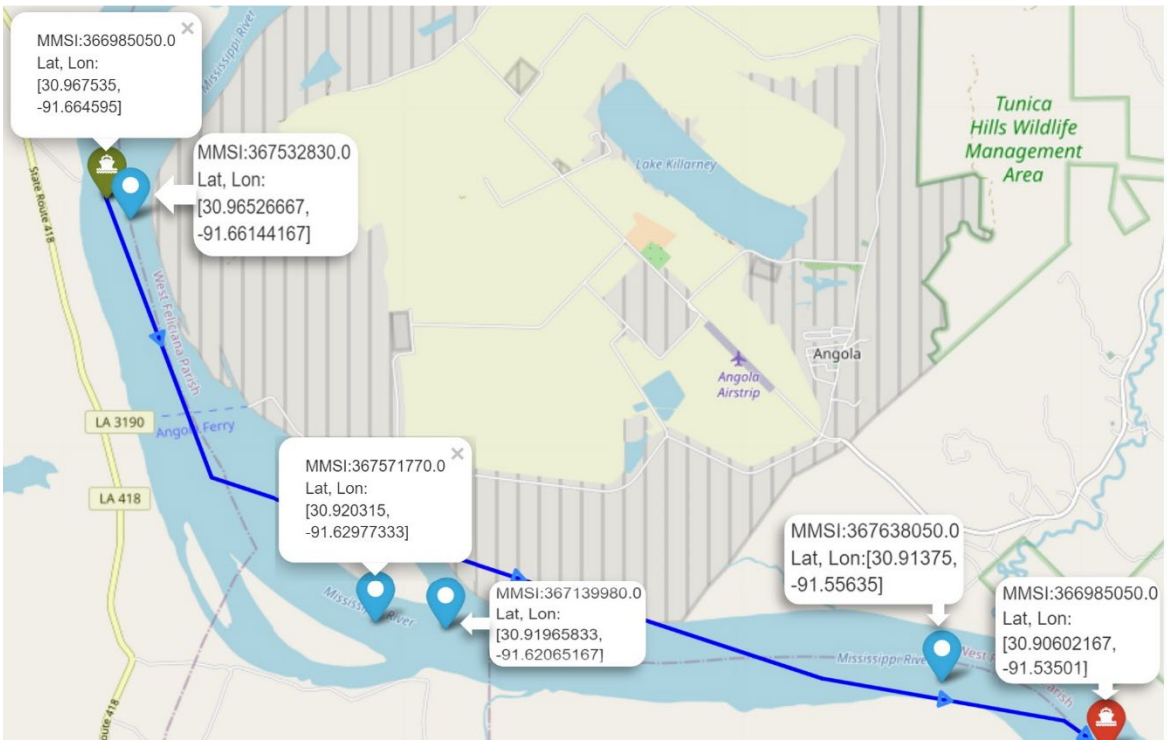


Figure 16: Trajectory of vessel 366985050.0 and other ships located in the span of its trajectory.

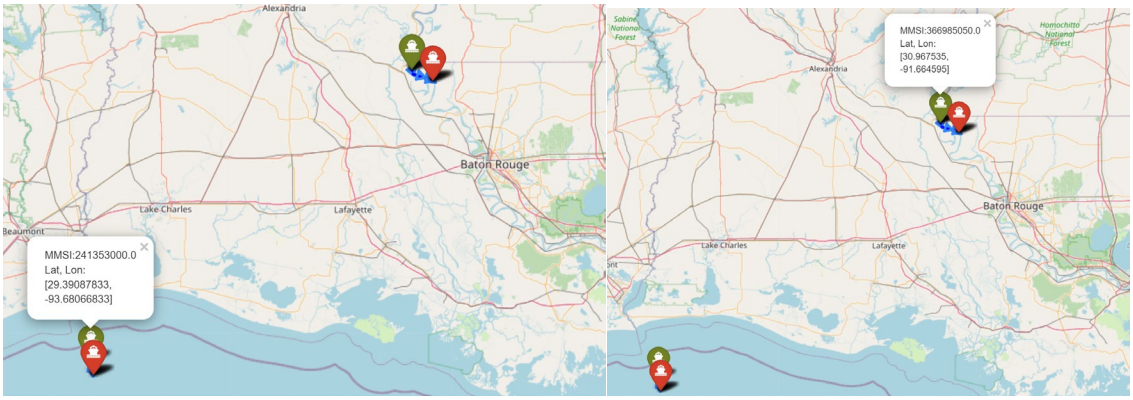


Figure 17: First two vessels in the subset of the AIS decoded data, showing in the Gulf of Mexico and the Mississippi River.

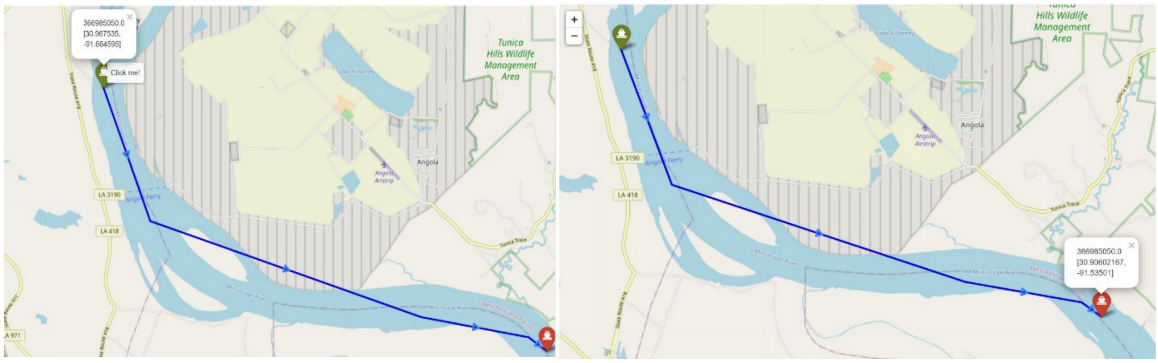


Figure 18: Zoomed Folium image showing second ship included in subset of AIS data traveling along the Mississippi River

Results:

In the above images, the green marker distinguishes the starting point of the maritime vessel, and the red marker is the ending point. When hovering over either the green or red marker, a popup will show that says "*click me*" (as seen in Figure 18 above). Once clicked, another popup displays the *MMSI*, ship unique identification number, along with the *latitude and longitude* of each ship as seen in Figure 17 and Figure 18. Any information that is desired to be added to the popup can be programmed to show there. Again, this is a subset of the obtained data and does not include the full trajectory of either of the vessels.

As a result of the target tracking added to the modified algorithms, we refer to the Modified DBSCAN Clustering Algorithm as the Modified DBSCAN Target Tracking algorithm and the Modified CURE algorithm as the Modified CURE for Target Tracking algorithm. Both are able to analyze big data to assist in elevating tracking and monitoring of maritime vessels by identifying individual vessels noted by their MMSI label with Geoscatter and MMSI label, latitude, longitude and trajectory with Folium in precise geographical areas.

4. Impacts/Benefits of Implementation (actual, not anticipated)

(i) Educational Impact:
The projected supported students in the Master's degree and Ph.D. degree in computational data enabled science and engineering (CDS&E) to undertake projects and dissertation research in the area of vehicular and marine traffic.

(ii) Increase enrollment in the CDS&E special topics on transportation data centric studies.
   a. CDSE 702 Special Topics in CDS&E- Data clustering analysis methods with AIS data.
   b. MATH 700 Topics in mathematics and statistics will applications to CDS&E- Linear Regression Methods and data Regularization with applications to Maritime and vehicular traffic data
   c. MATH 673 Quantitative Exploration of Data

(iii) Research field impact:
   a. The following publications have begun to receive citations in peer reviewed International Transportation Journals.
      i. Guojing Hu, Feng Wang, Weike Lu, Tor A. Kwembe, and Robert W. Whalin. (2020). A Cooperative

Bypassing Algorithm for Connected and Autonomous Vehicles in Mixed Traffic. IET Intelligent Transport Systems, 11pp. DOI: 10.1049/iet-its.2019.0707

ii. Guojing Hu, Feng Wang, Robert W. Whalin, and Tor A. Kwembe. (2020). Analytical Approximation for Macroscopic Fundamental Diagram of Urban Corridor with Mixed Human and Connected & Autonomous Traffic. IET Intell Transp Syst. 2021; 15:261–272.

iii. O. Osho, S. Hong and T. A. Kwembe, (2022). "Network Intrusion Detection System Using Principal Component Analysis Algorithm and Decision Tree Classifier," 2021 International Conference on Computational Science and Computational Intelligence (CSCI), 2021, pp. 273-279, doi: 10.1109/CSCI54926.2021.00117.

(iv) Stakeholder citations:

a. Hu, G., Wang, F., Lu, W., Kwembe, T.A., & Whalin, R.W. (2020). Cooperative bypassing algorithm for connected and autonomous vehicles in mixed traffic. IET Intelligent Transport Systems.

This paper has been cited at least 11 time in peer review journals.

b. Hu, G., Lu, W., Whalin, R.W., Wang, F., & Kwembe, T.A. (2020). Analytical approximation for macroscopic fundamental diagram of urban corridor with mixed human and connected and autonomous traffic. IET Intelligent Transport Systems.

This paper has been cited at least 10 times in peer review journals

(v). Presentations:

Mississippi Academy of Science (MAS)-2023, Biloxi, Mississippi

1. Demetric Baines, CDS&E Ph.D. student
   Topic: NUMERICAL SIMULATION OF THE MVG CONTROLLED HYPERSONIC FLOW. 87th Annual meeting of Mississippi Academy of Sciences, February 23-24, 2023, Biloxi, MS
2. Shimming Yuan, CDS&E Graduate Student
   Topic: A COMPARISON STUDY OF HIGHRESOLUTION FINITE DIFFERENCE SCHEMES FOR SUPERSONIC FLUID FLOW. 87th Annual meeting of Mississippi Academy of Sciences, February 23-24, 2023, Biloxi, MS

The 22nd International Conference on Information & Knowledge Engineering.

3. Morgan Smith, CDSE Ph.D. student
   Topic: Application of Machine Learning Classifiers Interfacing Google Colab and SKlearn to Intrusion Detection CSE-CIC IDS2017 Dataset.

(vi). Conference Presentations:

1. Smith, Morgan and Kwembe, Tor A. (2023). Application of Machine Learning Classifiers Interfacing Google Colab and SKlearn to Intrusion Detection CSE-CIC IDS2017 Dataset. The 22nd International Conference on Information & Knowledge Engineering
(IKE'23: July 24-27, 2023; Las Vegas, USA). CSCE 2023 BOOK of ABSTRACTS. ISBN # 1-60132-518-5; American Council on Science & Education / CSCE 2023. https://www.american-cse.org/csce2023/ . **Student Author: Morgan Smith**

2. Cheronika Manyfield-Donald, T. A. Kwembe and J. -R. C. Cheng, (2022). "A Modified Clustering Using Representatives to Enhance and Optimize Tracking and Monitoring of Maritime Traffic in Real-time Using Automatic Identification System Data," 2021 International Conference on Computational Science and Computational Intelligence (CSCI), 2021, pp. 285-289, doi: 10.1109/CSCI54926.2021.00119. **Student Author: Cheronika Manyfield-Donald**

3. O. Osho, S. Hong and T. A. Kwembe, (2022). "Network Intrusion Detection System Using Principal Component Analysis Algorithm and Decision Tree Classifier," 2021 International Conference on Computational Science and Computational Intelligence (CSCI), 2021, pp. 273-279, doi: 10.1109/CSCI54926.2021.00117. **Student Author: O. Osho**
4. H. Cotton and T. A. Kwembe, (2022). "Using Data Analytics to Forecast Violent Crime," 2021 International Conference on Computational Science and Computational Intelligence (CSCI), 2021, pp. 301-304, doi:10.1109/CSCI54926.2021.00122. **Student Author: H. Cotton**

5.Aljawfi, O.M., & Kwembe, T.A. (2022). Applying Machine Learning Algorithms to Identify the Associations Between Educational Background Factors and Problem-Solving in Technology-Rich Environment: An Investigation of Adult's Proficiency Level in PIAAC. Proceedings of the World Multi-Conference on Systemics, Cybernetics and Informatics. DOI:10.54808/wmsci2022.03.19. **Student Author: O. M. Aljawfi**

6. White, D., Howard, R., & Kwembe, T.A. (2021). MAKING PREDICTIONS TO INCREASE RETENTION AND GRADUATION RATES AT HBCUS WITH COMPUTATIONAL DATA ENABLED SCIENCE AND ENGINEERING (CDS&E) TOOLS IN HIGHER EDUCATION. EDULEARN21 Proceedings. DOI:10.21125/EDULEARN.2021.2562. **Student Authors: White, D and Howard, R.**

7. Howard, R., White, D., & Kwembe, T.A. (2021). UTILIZING CLUSTERING ALGORITHMS AND DATA ANALYTICS ON STUDENT ASSESSMENT DATA IN SECONDARY EDUCATION. EDULEARN21 Proceedings. DOI:10.21125/EDULEARN.2021.2552. **Student Authors: Howard, R. and White, D.**

8. White, D., & Kwembe, T.A. (2022). THE STEM/STEAM CONNECTION: AN ANALYSIS OF THE IMPACT OF STUDENT ENGAGEMENT AND SUCCESS USING DATA ANALYTICS METHODS OF HIERARCHICAL CLUSTERING ON PRINCIPAL COMPONENTS (HCPC). INTED2022 Proceedings. DOI:10.21125/inted.2022.2695. **Student Authors: White, D.**

9. Howard, R., White, D., & Kwembe, T.A. (2022). THE APPLICATION OF HIERARCHICAL CLUSTERING ON STUDENT ASSESSMENT DATA IN SECONDARY EDUCATION. INTED2022 Proceedings. DOI:10.21125/inted.2022.2721. **Student Authors: Howard, R. and White,D.**

(vii). List Computational Data Enabled Science and Engineering (CDS&E) Ph.D. students (first and last name) supported by this research project. Indicate gender and if they are a minority.
1. Eric S. Jackson (Male Black)
2. Lancelot Nelson (Male Black)
3. Ingrid Tchakoua (Female Black)
4. Demetric L. Baines (Female Black)

(viii). List CDS&E Ph.Ds. students supported by this grant that received their degree. Indicate gender and if they are a minority.
1. Cheronika Manyfield-Donald, Ph.D. in CDS&E (Female Black)
2. Oyeyemi Osho, Ph.D. in CDS&E (Male Black)

3. Di Wu (Female White) Has successfully defended her dissertation

5. Recommendations and Conclusions
   We recommend continued funding to enable the completion of the ongoing projects to complete the full development of an Automated Machine Learning (AutoML) model for monitoring and tracking maritime traffic for deployment and commercial use.