Hyperparameter-Optimierung bei Machine-Learning-Modellen in der maritimen Navigation

Hyperparameter Optimization on Machine Learning Models for Predicting the Vessel's Route Destination and Estimated Time of Arrival

Presented By:
Rajath Basawani Halgi.
Matriculation-Nr : 21756839
Course: Masters in Mechatronics

TM Moniruzzaman Sunny
Matriculation -Nr : 21757905
Course: Masters in Mechatronics

Jabir Mohamed Abdi
Matriculation -Nr : 21755680
Course: Masters in Mechatronics

Examiner: Prof. Dr.-Ing Carlos Jahn
Supervisor: Marvin Kastner, M.Sc.
Hamburg, 17. December 2020

Summary

AIS dataset comprises collection of information which relates to ship vessel. Every year thousands of ships travel across the world. Among them logistics ship plays a vital role in the economy of every country. Each country has ports as destination where typically ships reaches to load or unload goods. Time plays an important role in terms of logistics planning of the supply chain domain. The vessel reaching its destination before the assigned time or reaching the port after the scheduled time can cause various problems in terminal planning or transportation planning. To ensure good economic stability, every ship vessel must reach their destination on time. Also, the ports need to know or predict their arrival time. So that all the necessary transportation can be arranged in the time and unload. Each ship has planned path which is different for ports. In our project our primary focus is on the vessel's having Hamburg port as destination. In order to reach Hamburg port, the vessel can take two different routes namely Northern route and the Southern route. Our goals are divided into two tasks. Classification of the vessel's route coming either northern route or southern route to Hamburg. The other objective is to predict the Estimated Time of Arrival (ETA) for the vessel's reaching Hamburg port from those two routes. Machine Learning has been widely used to predict any continuous values such as share prices, weather etc. Prediction of ETA of the vessel is done with the help of regression-based ML model. The supervised machine learning can be used to predict the Route Identification with the help of Classification based ML models. The aim of the project is to determine the effect of Hyperparameter Optimization on machine learning model's performance metrices.

Gantt Chart

RESEARCH PROJECT WORK

Project start date: 3/5/2020

Scrolling Increment: 22

Milestone marker: 1

Milestone description	Assigned to	Progress	Start	No. days
DATA PREPROCESSING				
NORTHERN AND SOUTHERN ROUTE	SUNNY	100%	4/24/2020	14
ESTIMATED TIME OF ARRIVAL	SUNNY	100%	4/24/2020	14
CORRECTION ON PREPROCESSING	SUNNY	100%	6/2/2020	14
LITERATURE REVIEW				
RESEARCH OF ACADEMIC PAPERS	JABIR & RAJATH	100%	3/5/2020	
DOCUMENTATION	JABIR & RAJATH	100%	5/20/2020	14
TRAINING OF MODELS AND RESULTS				
PERFORMANCE METRIC SELECTION	ALL	100%	5/15/2020	14
TRAINING OF ALGORITHIMS	ALL	100%	5/24/2020	14
ERROR CORRECTION AND DOCUMENTAITON OF RESULTS	ALL	100%	6/1/2020	14
PROJECT REPORT AND PRESENTATION				
COMPILING OF PROJECT RESULTS	ALL	100%	6/20/2020	7
PRESENTATION OF THE FINAL RESULTS	ALL	100%	6/27/2020	7
ESTIMATED PROJECT COMPLETION			7/13/2020	

Table of Contents:

I. List of Abbreviations

AIS	Automatic Identification System.
ML	Machine Learning.
HPO	Hyperparameter Optimization.
TSS	Traffic Separation Scheme.
ST	Ship Type.
MMSI	Maritime Mobile Service Identity.
SOG	Speed Over Ground.
COG	Course Over Ground.
TH	True Heading.
ETA	Estimated Time of Arrival.
SMBO	Sequential Model based Global Optimization.
GP	Gaussian Process.
TPE	Tree-structured Parzen Estimator.
TPOT	Tree-based Pipeline Optimization Tool.
RMSE	Root Mean Squared Error.
TP	True Positive.
TN	True Negative.
FP	False Positive.
FN	False Negative

II. List of Figures

III. List of Tables

VI. List of Formulas

1	Root mean Squared error
2	Precision
3	Recall
4	Accuracy
5	F1- Score

1 Introduction

In this project a model is presented to predict the estimated time of arrival of the ship and the route identification of the ship using machine learning. Generally, in maritime domain Automatic Identification System (AIS) messages are used as a data source. The AIS messages have lot of data such as Course Over Ground (COG), Speed Over Ground (SOG), length of the ship, latitude, longitude, Draught of the ship, True Heading (TH), Call sign, Destination and Time. AIS is able to broadcast the vessel's location and status information. This data is collected at various places around the world and some of the locations for AIS data collections within Europe can be located at Kystverket, Hellenic coastguard, Dirkzwager and Marine Traffic [1]. AIS data are like the backbone for maritime transport. Based on this data, we can know how a ship has travelled, i.e. a ship can be seen visited one harbor and later visited other harbors.

In machine learning, the problem of choosing set of optimal hyperparameters for a learning algorithm is called Hyperparameter optimization or tuning [2]. ML models are parameterized so that their behavior can be tuned for a given problem using Hyper-Parameter Optimization (HPO). The ML models can have many parameters and finding the best combination of parameters is done by hyperparameter optimization [3]. They are used in process to help estimate model parameters. ML models can require different constraints, weights or learning rates to generalize different data patterns. The main approaches of HPO are Grid search, Random search, Bayesian Optimization, Gradient-Based Optimization and Tpot genetic algorithm. When it comes to HPO, Grid search has been widely used in many researches. Grid search will build a model on each parameter combination possible. While as the random search will select a combination of parameters randomly. Selecting the right choice of hyperparameter values will affect the performance of a machine learning model. In this project the chosen Hyperparameter Optimization methods are Random search, Bayesian optimization and Tpot genetic algorithm. The work of Hyperparameter Optimization algoirthms is not only to optimize over the variables which are continuous or discrete but it must choose which variables to optimize at the same time [4]

1.1 Research Questions and Approach

The project mainly has three research questions namely:

- Literature review on Hyperparameter Optimization Algorithms.
- What are the performances of a selected choice of Hyperparameter Optimization algorithms in machine learning models used for predicting the estimated time of arrival of the vessel?
- What are the performances of a selected choice of Hyperparameter Optimization algorithms in machine learning models used for predicting the route navigation of the vessel?

The normal approach for any ML model will be searching for datasets which has AIS data from various dataset websites like Kaggle, Socrata, Google public datasets etc. Since the required dataset is provided by Institute of Maritime Logistics department, we can skip the dataset searching part and jump right into the Preprocessing part. As the given data will be having large number of unwanted data, preprocessing of the raw dataset is required. The main aim of preprocessing is to optimize the dataset and to stay within the experiment range. After the preprocessing is completed the next step is to determine the features and labels for ML model. Searching for regression-based algorithms to predict the Estimated Time of Arrival (ETA) of the vessel and different classification algorithms to predict the vessel navigation, i.e. to predict whether the vessel has taken northern route or the southern route to arrive at Hamburg port.

Literature review on Hyperparameter optimization is done to find out any new Hyperparameter optimization algorithms currently being used and with the help of recent articles on HPO to find out how different HPO algorithms react to different kind of datasets. The idea of literature review is to find latest research articles on HPO from websites like TUHH Library and Google Scholar. Shortlisting a list of HPO algorithms and applying these on the regression and classification algorithms. Applying various Hyperparameter optimization algorithms on the predicted model to increase its performance. Comparing different HPO methods to observe the performances of the ML model.

2 Literature Review

2.1 Literature Review Process

There are mainly five steps in conducting a systematic literature review by [5] and they are:

1 – Scoping.

2 – Planning.

3 – Identification.

4 – Screening.

5 – Eligibility

Scoping is an exploratory process of assessing the literature for its likely quantity and quality [5]. Do we have a clear idea of the type of research finding that will be relevant to addressing your research questions? [5] Methodologically it helps in determining appropriate review methods and logistically it assists in estimating how much time, money and reviewer effort will need to be expended for a specific review [5]. As the research question was already defined as “what are the performances of a selected choice of Hyperparameter optimization algorithms in machine learning models used for predicting the estimated time of arrival and the route navigation of ships?” and “How can we measure the performances of these Hyperparameter Optimization algorithms?” We already knew what we want to know and what topics.

In Planning, The research question needs to be broken down into individual concepts to create the search terms to find relevant article [5]. Thinking of alternative terms and concepts which address the same question as it is common for a range of terms to be used to describe the same phenomenon or research area and things to consider are Synonyms, Singular/Plural forms, Different spellings [5]. Breaking the research question into individual concepts like Hyperparameter optimization, estimated time of arrival, maritime, Navigation, Route Identification, Machine Learning Models. Instead of using Estimated time of Arrival,

using ETA as a search term etc. Synonyms like Hyper-Parameter, Optimization, Rise, Increment, Inflation, Development, Breakthrough, Design, Innovation, Method, Result, Navigate. Trying different spellings like marinetime, optimisation etc.

In Identification, making use of the search terms to search at least two different electronic databases [5]. The aim is now to find all available published and unpublished work which addresses the research questions, operationalized through your search terms [5]. Electronic databases used to search articles on HPO algorithms are TU library database, web of science, Scopus, Emerald Insight, Google Scholar. Journals and books used are “Automated Machine Learning”, Journal of Machine learning.

In Screening, reading the Title and Abstract of all work identified by the searches. Some may pass the inclusion and exclusion criteria, we will need to obtain the full-text version and read that carefully [5]. First reading the article’s abstract, if it contains the terms “Hyper parameter Optimization algorithms” then we select the article.

In Eligibility The focus now shifts from sensitivity to specificity and we now need to sift the full-text version of potentially eligible articles to see if each is indeed suitable for inclusion [5]. The main objective of the literature review in this project is find out various Hyperparameter optimization algorithms used in recent times and how different HPO algorithms affect different regression and classification algorithms.

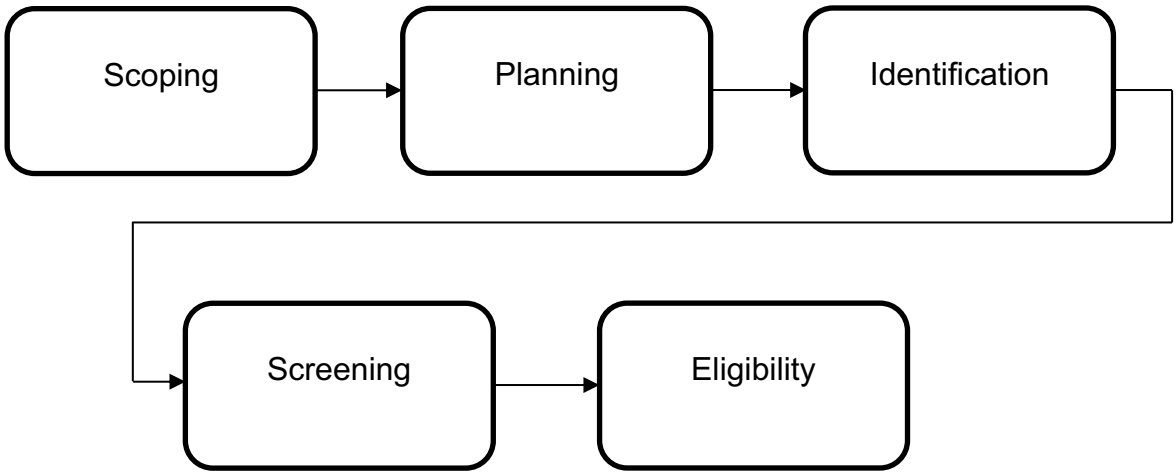

Figure 1 : Literature review process flowchart

2.2 Literature Review on Hyperparameter Optimization

As the Hyper parameter optimization is an important stage in any Machine Learning model, a research article titled “Algorithms for Hyper-Parameter Optimization” [4] provides an insight on various HPO algorithms. The author has described some of the HPO algorithms namely Sequential Model-based Global Optimization (SMBO), The Gaussian Process Approach (GP), Tree-structured Parzen Estimator Approach (TPE), Random Search, Sequential Search. SMBO is used in plenty of applications where the evaluation of the fitness function is expensive [6] whereas GP has been recognized as a good method for modeling loss functions in model-based optimization [7]. The author concludes Random Search has been seen to be sufficiently efficient for a ML model compared to SMBO, GP, TPE and Sequential Search. The same author has another article called “Random Search for Hyper-Parameter Optimization” [8] in which the author has compared Random Search with other HPO algorithms such as Grid Search and Sequential Manual Optimization and has said that random experiments are more efficient because not all hyperparameters are equally important to tune. Grid search experiments allocate too many trials to the exploration of dimensions that do not matter and suffer from poor coverage in dimensions that are important. The computational time for Random Search is less than that of Grid Search.

A recent article titled “TPOT-SH: a Faster Optimization Algorithm to Solve the AutoML Problem on Large Datasets” by [9] published in 2019 reveals Tree-based Pipeline Optimization Tool (TPOT) genetic algorithm as HPO algorithm for ML models. The hyperparameters of TPOT genetic algorithm can be seen as Population size, Generations, Mutation, Crossover, Candidate evaluation and Number of jobs. The author has stated that this algorithm is considered to be one of the fastest optimization algorithms when compared to Random Search and Grid Search particularly when there are large datasets.

A Book on various HPO algorithms by [2] is titled as “Automated Machine Learning”. The author has showcased many HPO algorithms such as Grid Search, Random Search, SMBO, Bayesian Optimization, TPOT, Hyperopt etc. Apart from Hyperopt we have seen all other HPO algorithms. In the book Hyperopt is stated that if the

user has to use this algorithm then he has to define three important steps namely A search Domain, An objective function and an optimization algorithm.

As we are going to focus on Random Forest Regression Algorithm in this project. An article titled “Hyperparameters, Tuning and Meta-Learning for Random Forest and Other Machine Learning Algorithms” by [3] published in year 2019 has valuable insights on hyperparameters focusing on Random Forest algorithm. The author has stated that the hyperparameters that needs to be set in Random Forest are the number of variables that are regarded in each split. The HPO algorithms used in this research article are Random Search and Grid Search. The hyperparameters of random forest that needs to be tuned are mtry, Sample size, Replacement, Node size, Number of Trees and Splitting rule. The author has discussed about several hyperparameters influence on the performance of Random forest algorithm such as by setting the number of trees as high meaning higher the number trees tend to yield better results in terms of performance and precision of variable importance’s [3]. Normally Random forest is known to provide good results in default settings [10]. Some of the hyperparameters have minor influence on the performance on ML model such as sample size and node size.

An article” Tunability: Importance of Hyperparameters of Machine Learning Algorithms” [11] shows the importance of HPO on ML models and the author has said that the Modern supervised machine learning algorithms involve hyperparameters that have to be set before running them. The author Probst has mentioned that a tuning strategy for hyperparameters in a ML model is an important stage and in addition to picking up an efficient tuning strategy the user has to determine the corresponding ranges, scales and potential prior distributions for subsequent sampling [11].The selection of HPO algorithms in this project is based according to the frequency of the HPO algorithms used in most of the research article journals and books.

The selected HPO algorithms are:

- Random Search.
- Bayesian Optimization.
- TPOT genetic algorithm.

3 Preprocessing of Dataset

Preprocessing is an important stage in any machine learning model. The process of removing unwanted data and to set the definite labels and features for a machine learning model is called preprocessing. The raw AIS data has characteristics such as MMSI number, Ship type, Length, Breadth, COG, SOG, Draught, Longitude, Latitude, TH, Destination, Name, Callsign, Time. The definitions of AIS data is shown below.

3.1 Definitions

In the following, the definitions of [1,2] are used:

MMSI number: Every vessel will be having a unique number for identification known as MMSI number. It stands for Maritime Mobile Service Identity (MMSI). In general, it contains of nine digits. It is sent over a digital form over a radio frequency channel in order to uniquely identify ship stations, ship earth stations, coast stations, coast earth stations, and group calls.

Ship type: There are different types of ships on the ocean. There is a specific two digits for different types of ships. Some of the types are container ships, bulk carriers, oil tankers, gas carriers etc.

Length: Different vessels will be having different length according to the storage capacity of the vessel.

COG: Course Over Ground is the actual direction of progress of a vessel between two points with respect to the surface of the earth. COG may differ from various factors such as tide, wind effect and currents etc.

SOG: Speed Over Ground is the speed of the vessel relative to the surface of the earth. Speed over ground includes current forecast whereas the course over ground is for the next waypoint.

Draught: It is the vertical distance between the waterline and the bottom of the keel. It determines the minimum depth of water a vessel can safely navigate.

Latitude: It is the measurement of a part of the Earth in relation to the north or south of the Earth's equator. The latitude position of the ship is recorded everytime change is detected.

Longitude: It is a geographic coordinate that specifies the east-west position of a point on the earth's surface. Based on latitude and longitude the exact location of the ship can be traced.

True Heading: The heading refers to the direction of the vessel pointing. It is typically based on compass directions, so 0° indicates a direction towards true north. 90° indicates true east, 180° is for true south and 270° is for true south.

Call-signs They are assigned as unique identifiers to ships and boats. All radio transmissions are individually identified by the call sign.

Time For each location ie the latitude and longitude, COG, SOG, TH etc is recorded with respect to time. So that at given point of time all the charecterstics of the ship can be found.

3.2 Preprocessing

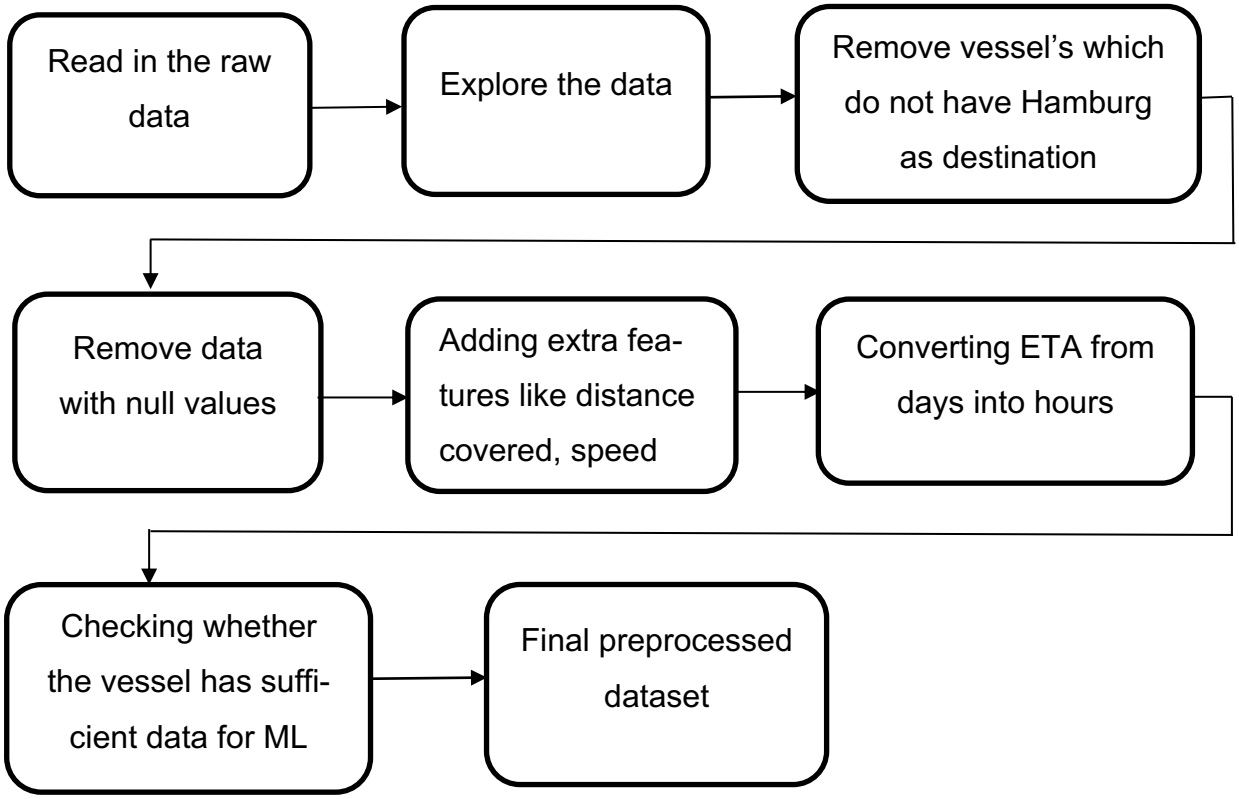

Figure 2 : Preprocessing Flowchart

The dataset is provided by Institute of Maritime Logistics department. The raw data has lot of unwanted data which needs to be filtered out. The first few rows of the dataset before the preprocessing is shown below.

ID	ST	L (m)	B (m)	D (m)	Longitude	Latitude	SOG (kn)	COG (kn)	TH (deg)	Destination	Name	Callsign	time_UTC
V1	0	162	25	6,3	5.517354	53.96507	16.3	48	46	SE Gothenburg	valentine	ONJB	07:18.2
V1	0	162	25	6,3	5.523613	53.96833	16.2	49	46	SE Gothenburg	valentine	ONJB	08:24.3
V1	0	162	25	6,3	5.529877	53.97159	16.2	48	45	SE Gothenburg	valentine	ONJB	09:30.1
V1	0	162	25	6,3	5.535505	53.97463	16.1	48	46	SE Gothenburg	valentine	ONJB	10:30.2
V1	0	162	25	6,3	5.541227	53.97757	16.2	49	45	SE Gothenburg	valentine	ONJB	11:30.2
V1	0	162	25	6,3	5.546265	53.98022	15.9	47	46	SE Gothenburg	valentine	ONJB	12:24.3
V1	0	162	25	6,3	5.552474	53.98339	15.8	47	45	SE Gothenburg	valentine	ONJB	13:30.1
V1	0	162	25	6,3	5.558077	53.98634	16.1	47	44	SE Gothenburg	valentine	ONJB	14:30.0
V1	0	162	25	6,3	5.563687	53.98936	16.1	47	45	SE Gothenburg	valentine	ONJB	15:29.9
V1	0	162	25	6,3	5.569355	53.99234	16.1	48	45	SE Gothenburg	valentine	ONJB	16:30.0

Table 1 : Raw Dataset

In the above table ST represents Ship type, L is Length, B is Breadth and D is draught. As seen in the raw data, there are lot of vessel's which travel to different destinations other than Hamburg. So filtering according to the vessel's destination which has Hamburg as destination and removing all the other vessel's which do not have Hamburg as destination. As different vessel's will be having unique ID, Counting the total number of vessels in the dataset id done using ID. Once the total number of vessels known, Creating a subset for each indiviual ships is done in jupyter notebook. As there will be vessel's which start at Hamburg port needs to be filtered. Keeping in mind that there might be vessel's whose departure is from Hamburg port also needs to be removed. As the raw data might have same rows needs to be taken care of by removing those repeated rows. Exploring the dataset and trying to find out which all attributes has an influence on ETA of the vessel and to identify the route of the vessel. And to remove all the attributes which does not have any influence on ETA prediction and Route identification.

3.3 Traffic Separation Schemes

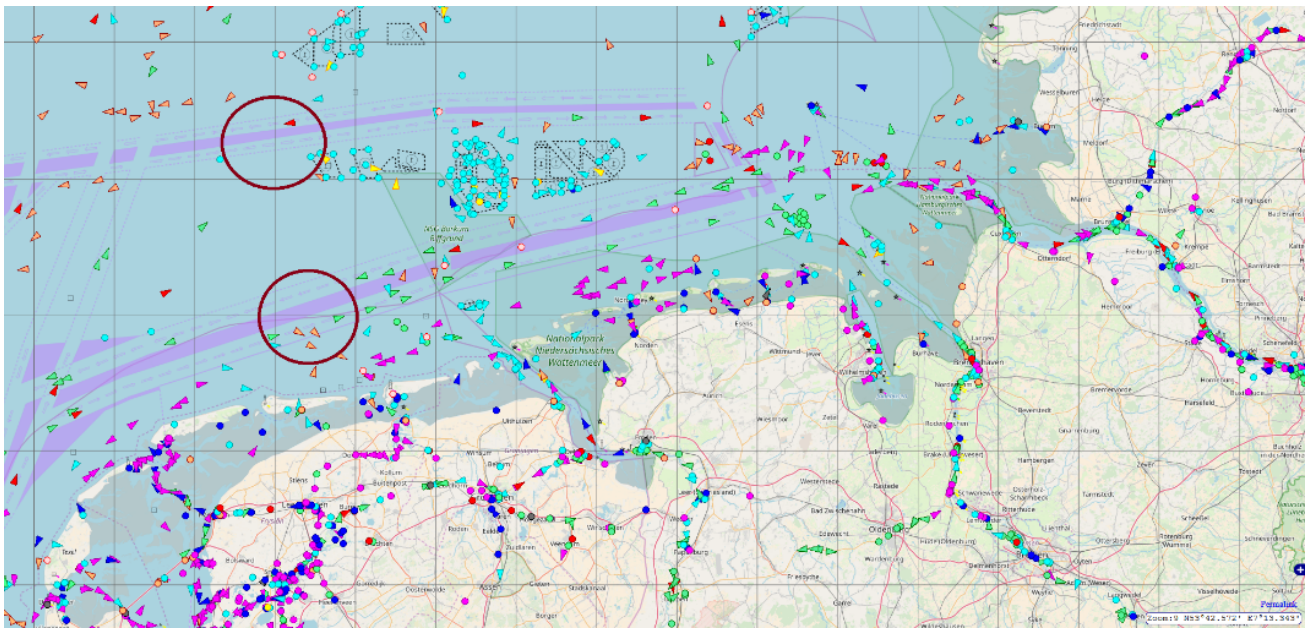

Figure 3: Traffic Separation Scheme (Image Source: http://map.openseamap.org/)

Traffic Separation Scheme (TSS) can be described as a traffic-management route-system ruled by the IMO where the traffic lanes indicate the general direction of the traffic flow [13] . TSS helps the pilot of the vessel for an efficient and secure navigation and it contributes to the safety of life at sea [13].

The vessels can take two routes in order to reach Hamburg port. The two routes have violet colored lanes. The Brown circles indicates the polygon through which the data has been extracted through SQL. If the vessel goes through the top brown circle then the vessel's direction is noted as Northern route whereas if the vessel passes through the bottom brown circle in the figure 3 is noted as the Southern route. The small triangles and circles with different color represents the Marine Traffic. From Figure 3, it can be seen that if a vessel passes through northern route to Hamburg the vessel might take long time to reach Hamburg port when compared to a vessel taking southern route to Hamburg port. The Southern route is shorter when compared to the northern route, this might cause the vessel taking this route to Hamburg port can reach the destination earlier. The northern and southern polygon are discussed in next section.

3.4 Polygons

There are two polygons namely Northern and Southern for the route identification. One more polygon is at the Hamburg port and is known as Hamburg port polygon. If the vessel has passed through Northern polygon and entered Hamburg port, The vessel's route is identified as Northern route. Where as if the vessel passes through Southern Polygon, the vessel's route is considered as Southern route. The Northern , Southern and Hamburg port polygon can be seen in figure 4.

Using the co-ordinates of Northern, Southern and Hamburg polygon and with the help of libraries in jupyter notebook is done to map the polygons.

The polygons have been numbered and it can be represented as shown below.

1 – Northern Polygon.

2 – Southern Polygon.

3 – Hamburg Polygon.

Figure 4 : Polygons

First thing to consider is the amount of data each vessel has. Taking into consideration if and only if each vessel's data containing a minimum of 100 dataset. So, filtering out the vessels which does not suffice the minimum data to model. There are two csv files, one for the vessels following Northern route and one for the vessels following Southern route.

There are in total 113 ships in the northern csv file. In order to get the unique Hamburg ships using ID and arranging the data sequentially according to their total data points. The best way to filter only the Hamburg ships is by making a for loop by taking shipindex in the range. The ships which pass through the northern polygon and goes through Hamburg polygon are the only ones to be considered. The data before the northern polygon and the data after the Hamburg polygon needs to be discarded. Checking how many points has intersected with the northern polygon and note that indexing is created according to the original data index, so the first data that intersects first in the north polygon is done by using the function intersections_northern.iloc[0].

Since there will be vessels which originate from Hamburg and passes through northern polygon needs to be discarded. This can be done by creating if else loop. As we know the time at which the vessel intersects northern and the Hamburg polygon from the dataset, creating a separate csv file for each vessel using the loop and the time at which it intersected the northern and the time at which it has intersected Hamburg polygon. The loop creates a file if the time of intersection at northern is before than the Hamburg polygon. If the time of intersection at northern is after Hamburg polygon then the vessel has been discarded.

The files are made if and only if the number of data found in columns are above hundred and the direction of the vessel must pass through northern polygon and towards Hamburg port polygon. In some cases, even if the data is sufficient yet the file is not made as the vessel will be going from Hamburg port polygon and then through northern polygon. All the separate csv files will be merged into a single csv

file which would contain all the vessel's which has the destination of Hamburg port and passed through the northern polygon.
The Southern file contains all the vessels which has passed through the southern polygon. This file has lot of data since the number of ships are around 460. The same process is repeated for the southern vessel's as the southern vessels. As we know the time at which the vessel intersects southern and the Hamburg polygon from the dataset, creating a separate csv file for each vessel using the loop and the time at which it intersected the southern polygon and the time at which it has intersected Hamburg polygon. The loop creates a file if the time of intersection at southern is before than the Hamburg polygon. If the time of intersection at southern polygon is after Hamburg polygon then the vessel has been discarded. From this we will get a large number of separate csv files for the vessels taking the southern polygon route to Hamburg port. Making files is the result of conditions that the data has a minimum of 100 datapoints (for better ML result) and if and only if the time of the vessel when it reaches Southern polygon must be before it reaches the Hamburg port polygon. If these two conditions satisfy, then a separate csv file will be created containing a vessel travelling from Southern to Hamburg port. Combining all the csv files into a single file.

3.5 Data Cleaning

Now that there are two separate csv files containing vessels traveling from the northern polygon to Hamburg port and southern polygon to Hamburg ships. There are certain columns which might contains Nan values. These data's can create errors during computing operations. In order prevent any errors for computing, any row containing Nan or NUII values needs to be removed. There can be a possibility of some duplicate rows and this can affect the accuracy of the ML model so these duplicate rows have to be removed as well.

Adding a feature column can improve training of ML model. Therefore, adding an extra features column called 'Distance Remaining' and 'Speed' can improve the training. Adding one more feature called Bool, which represents Boolean. In this column, if a vessel has taken Northern route to Hamburg port then that vessel is

represented as 1 and if the vessel has taken Southern route then that vessel will be having 0. To calculate the exact earth distance is done with the help of libraries. As the time is stated in days, converting the days into hours so that there won't be any error in prediction of estimated time of arrival of the vessel. It calculates the time taken in hours by the ship to reach the final destination point from its current point. This feature is one of the essential features to calculate the ETA of the vessel.

The vessel's which does not have a minimum of 100 data needs to be removed. The features which did not have any effect or feeble effects are Name, Destination, True Heading, Call sign, Ship Type have been removed. As the machine does not understand Texts. The preprocessed dataset contains ID, Length, Breadth, Draught, Latitude, Longitude, SOG, Boolean, Distance Covered and ETA. Now, the dataset has been preprocessed. The below table 2 represents the sample of the preprocessed dataset of vessel having an ID V5 and taking the Northern Route to Hamburg port.

ID	L (m)	B (m)	D (m)	Longitude	Latitude	SOG (kn)	Speed (km/h)	Distance Remaining (km)	**ETA (h)**	**Boolean**
V5	152	24	6,8	5.838777	53.63944	14,3	34.73	440.630956	12.685507	1
V5	152	24	6,8	5.845800	53.64076	14,3	34.72	439.857752	12.667177	1
V5	152	24	6,8	5.852149	53.64198	14,2	34.71	439.158871	12.650496	1
V5	152	24	6,8	5.858515	53.64320	14,2	34.71	438.458162	12.633836	1
V5	152	24	6,8	5.864876	53.64441	14,3	34.70	437.758031	12.617179	1
V5	152	24	6,8	5.869312	53.64621	14,3	34.69	436.953012	12.598155	1
V5	152	24	6,8	5.875213	53.64798	14,3	34.68	436.153718	12.576023	1
V5	152	24	6,8	5.879932	53.64912	14,3	34.68	435.785301	12.558014	1
V5	152	24	6,8	5.885789	53.65098	14,3	34.67	434.875519	12.539571	1
V5	152	24	6,8	5.889946	53.65231	14,3	34.67	434.139942	12.513047	1

Table 1 : Preprocessed Dataset

The units of measurement for the above attributes are:

- Length of the ship in meters (m).
- Breadth of the ship in meters (m).
- Draught of the ship in meters (m).
- SOG in Knot (kn).
- Distance Remaining in Kilometer (km).
- Speed in Kilometer per Hour (km/h).
- ETA in hours (h).

As the raw data has been preprocessed, taking one ship from southern route and one vessel from northern route and plotting the Heatmap of the vessel's direction using libraries. To plot the heatmap the only attributes required are ID, Latitude and longitude. So, dropping all other attributes such as Length, Breadth, SOG, COG, Distance Remaining, ETA for the visualization of the Heatmap is achieved. The Heatmap is used to visualize the direction of the vessel and to verify if the dataset is completely preprocessed for ML model.

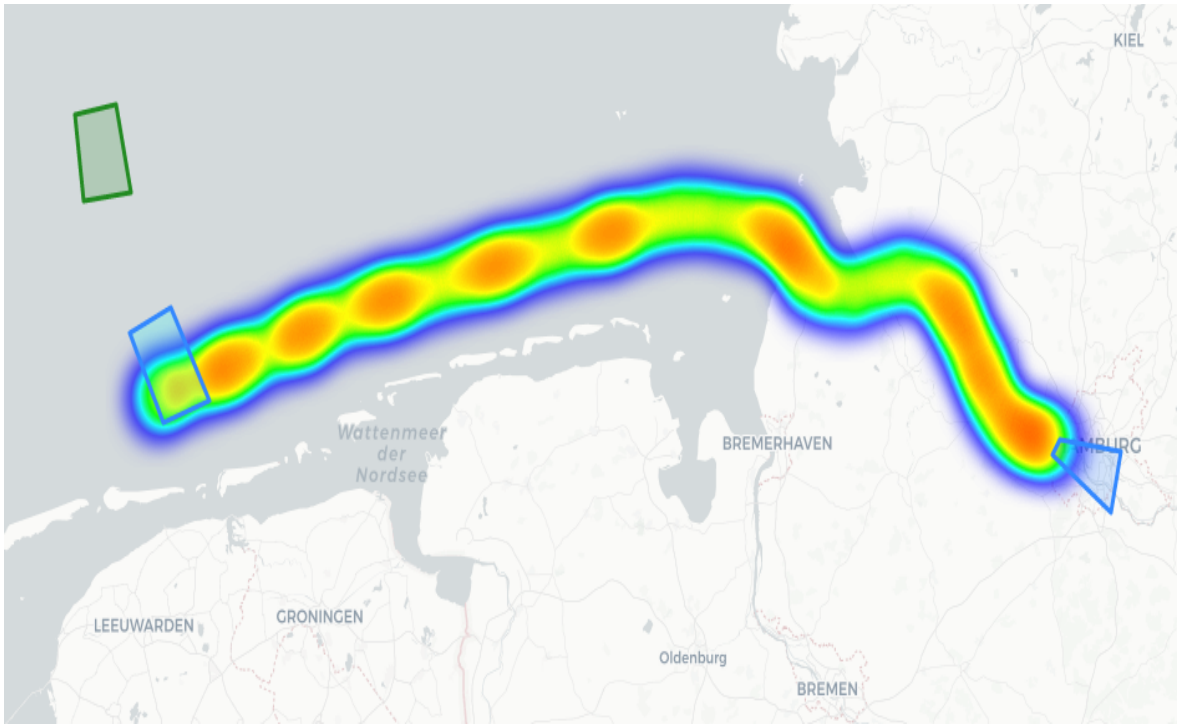

Figure 5 : Vessel route from Southern polygon to Hamburg polygon

Figure 5 shows the direction of the vessel from the southern polygon to the Hamburg port polygon. It can be seen from the figure, the data before the southern polygon has been removed as well as the data of the vessel after the Hamburg port polygon is removed through preprocessing. There might be vessel's which can have a stop at Hamburg port and then might continue the journey towards next destination so, the data after Hamburg port is removed to find ETA of any vessel which arrives at Hamburg port.

Figure 6 represents the vessel which has taken Northern route to reach Hamburg port can be seen. Now that the preprocessing of the dataset is completed, the further steps are to find a regression algorithm for ETA prediction of the vessel and a classification algorithm for the vessel's route identification and to apply HPO algorithms on these algorithms.

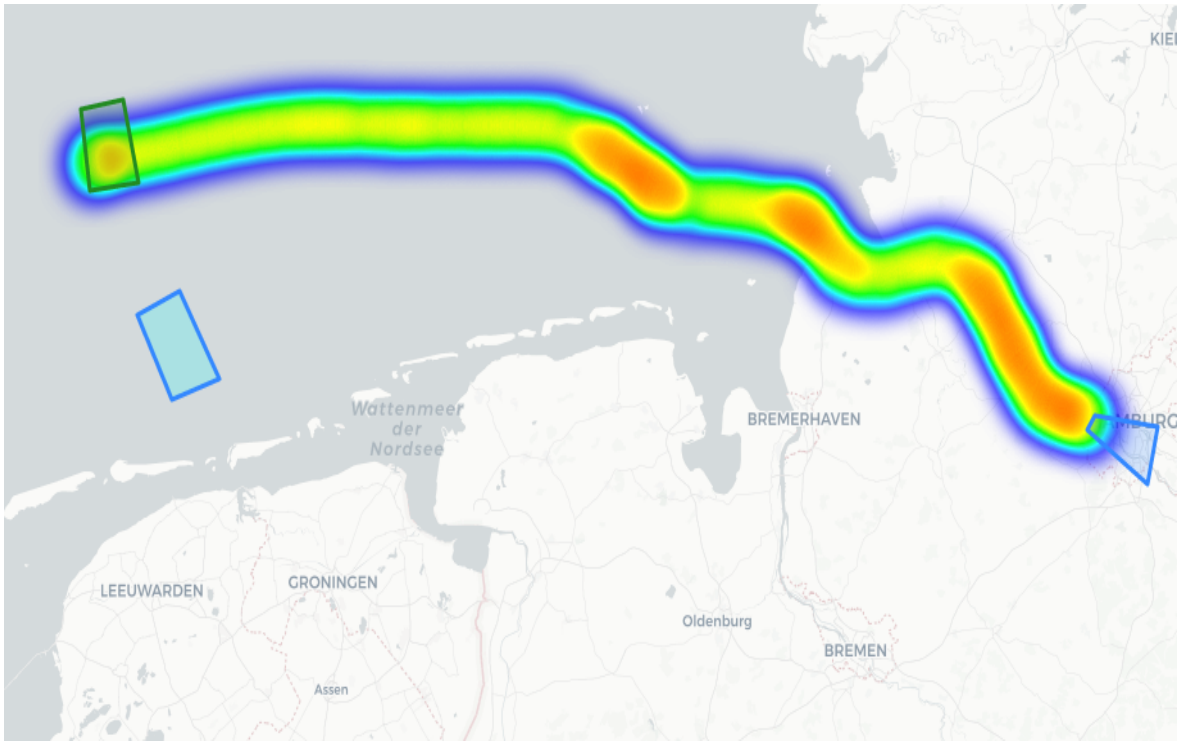

Figure 6 : Vessel route from Northern polygon to Hamburg polygon

4 Methodology

To predict ETA of a vessel, Regression algorithm is required as ETA will be having continuous values. Whereas the Route Identification of the vessel requires classification-based algorithm as the route is either North or South. The selection of Regression based algorithms for ETA and Classification based algorithms for route identification is discussed in this section.

4.1 Algorithm Selection

There are many Regression algorithms to predict continuous values in ML. Some of the common regression algorithms are Linear Regression, Naïve Bayes, Random Forest Regression, K Nearest Neighbor (KNN), Support Vector Machine (SVM) etc. To predict the ETA of the vessel is done with the help of Random Forest Regression in this project. Furthermore, Random Forest regressor has produced good results particularly on large datasets [3]. The Route identification of the vessel is done with the help of Logistic Regression in this project.

4.1.1 Random Forest Regressor

Random forest is a Supervised Learning algorithm which is used for classification and regression. There is a limit to some percentage of total split of the number of features on each node which is hyperparameters [3]. This is to make sure that the model does not rely heavily on any individual feature, so it makes use of all features. Each tree selects a random sample from the dataset when it splits. This is how Random forest avoids overfitting [3].

Advantages of Random Forest Regressor are:

- It runs efficiently on Large datasets.
- It maintains good accuracy even though some of the data's are missing as it has an effective method for estimating missing data.
- It is one of the most accurate learning algorithms.
- It has low bias and moderate variance

The Hyperparameters of Random Forest [3] are shown below

Hyperparameter	Description
Number of trees	Number of trees in the forest
Sample size	Number of observations that are drawn for each tree
Node Size	Minimum number of observations in a terminal node
mtry	Number of drawn candidate variables in each split
Splitting rule	Splitting criteria in nodes.

Table 3: Hyperparameters of Random Forest Regressor

4.1.2 Logistic Regression

To predict the Route identification of the vessel (Northern route or Southern route) is done by Logistic Regression. It is used when the dependent variable is binary i.e. 0 or 1, True or False, yes or no. To predict the route identification of a vessel a label named Boolean is used. The label Boolean has only two variables namely 0 and 1. Where 1 represents the Northern route and 0 represents Southern route.

There are two types of logistic regression:

- Binary classification.
- Multi Linear classification.

Some of the advantages of Logistic regression are:

- It is easier to implement, interpret and efficient to train.
- It can be easily extended to multiple classes.
- It is very fast at classifying unknown records.
- It has a good accuracy for many simple data.

4.1.3 Hyperparameter Optimization Algorithms

The selection of HPO algorithms is done from the literature review on HPO algorithms. The selected HPO algorithms for ETA prediction and Route Identification are as follows

- Random Search.
- Bayesian Optimization.
- Tpot Genetic Algorithm.

Random Search Algorithm:

Random search is an alternative for Grid search. It does not make any assumptions on the machine learning algorithm being optimized [2]. It samples configurations of hyperparameters of a machine learning at random and evaluates their performances until a certain budget of the search is exhausted i.e. number of trials, performance criteria [2]. This has an advantage of having a useful baseline since it makes no assumption of the machine learning algorithm and could achieve their optimal performance. The main issue for random search is that takes quite long but not as long as grid search method to obtain optimal performance.

Bayesian Optimization Algorithm:

Bayesian optimization is the current state of the art of optimization for global optimization. It is currently used in tuning deep neural network algorithms used for image classification, speech recognition and neural language modeling [2].

Bayesian optimization is an iterative learning algorithm. The two main components of the algorithm are it has is the surrogate model and an acquisition function. The surrogate model employs a gaussian process because it has a closed form computability of the predictive distribution [14]. In each iteration, the surrogate model is fitted to all the observations of the target function made so far. Then the acquisition function, which uses the predictive distribution of the probabilistic model, determines the utility of different candidate points, trading off exploration and exploitation. The acquisition function is cheap to compute and can achieve optimization of the machine learning model.

TPOT Genetic Algorithm:

This is a population-based algorithm that has a population (for this case a set of machine learning models with different hyperparameter setting). The algorithm then generates a new generation by making combinations of different hyperparameters and evaluating their performances [2]. The algorithm has an advantage of running in parallel since a population can have different members being evaluated in parallel [15].

As the HPO algorithms (3 HPO) has been chosen and with the help of Random Forest Regressor, the prediction of ETA of the vessel needs to be done. For Prediction of the Route Identification is done with the help of Logistic Regression.

Further Steps after the selection of algorithms are

For ETA prediction:

- Setting up experimental setup for ETA prediction.
- Applying Random Forest Regressor for ETA prediction.
- Comparing the training Duration of HPO algorithms
- Tuning of the HPO parameters.
- Comparing the performance matrices with three different HPO algorithms and without the HPO algorithm.

For Route Identification:

- Setting up experimental setup for Route Identification.
- Applying Logistic Regression for Route Identification.
- Comparing the training Duration of HPO algorithms.
- Tuning of the HPO parameters.
- Comparing the performance matrices with three different HPO algorithms and without the HPO algorithm.

4.2 ETA prediction

For ETA training, the preprocessed data is used for the training experiment. In order to avoid repetitive work one specific csv file is made. For every training the same preprocessed csv file is used to split the dataset for training and testing. As discussed before, Random forest regressor is chosen for the ETA prediction of the vessel. The prediction is done without the HPO and then ETA prediction is done with the three HPO algorithms to compare the effect of HPO algorithms on Random Forest regressor. The idea of applying HPO algorithms to tune to get a better result.

Due to limitation of the computing power, the optimization of the experiment is done in a way such that it does not face any memory leaking issues. The main constraint of the experiment here is the large dataset. As the dataset is huge to compute any simple wrong experiment will keep running the experiments forever and the jupyter notebook gets crashed. After many attempts of running the experiments in a proper way, it is found that certain range must be set for tuning the HPO. Setting higher parameter value tends to take higher training time and higher parameter value also results in memory leaking and the experiments prone to fail. The optimal parameter values are found based on trial and error method.

ETA Experiment setup:

1. **Hyperparameter Optimization to be tested:**
    - Default values of the ML model (No tuning).
    - Random Search HPO algorithm.
    - Bayesian Search HPO algorithm.
    - Tpot genetic Algorithm.

2. **Machine Learning algorithm used:**
    - Random Forest Regressor.
    - Number of iterations of testing the performance is 5.

3. **Performance Metrics:**
    - Mean Squared error (Hours)
    - Duration (Seconds).

4. **Hyperparameters tuned and their 'search space':**
    - Maximum depth: Range between 2-50
    - Maximum features: 'Auto'
    - Minimum samples leaf: Range between 2-20
    - Minimum samples split: Range between 2-20
    - Number of estimators: Range between 1-100

A total of 5 tests have been made to predict the ETA of the vessel. The Hyperparameters tuned for Random search and Bayesian search is the 'N_estimators' parameter for five different tasks. Whereas for the Tpot genetic algorithm 'Population Size' parameter is tuned for different tasks. The hyperparameters tuned for the ETA prediction in all the 5 tasks are shown below

Test number	N estimators for Random and Bayesian	Generation Size for Tpot algorithm
1	10	15
2	20	5
3	30	30
4	40	20
5	100	30

Table 4: Tuning Hyperparameters for ETA prediction

The performance metrics is Root Mean Squared Error. RMSE is the standard deviation of the errors which occur when a prediction is made on a dataset. This is the same as MSE (Mean Squared Error) but the root of the value is considered while determining the accuracy of the model.

$$\text{RMSE} = \sqrt{\sum_{i=0}^{n} \frac{(\hat{y}_i - y_i)^2}{n}} \qquad \text{...(1)}$$

$\hat{y}_i$ are the predicted values.

y_i are the observed values.

n is the number of observations.

4.3 Route Identification

To predict the vessel's route, a specific csv file has been produced to avoid the repetitive work. A label named 'bool' is introduced for the route identification. Logistic regression is used to predict the route identification. The route is either North or South, so the binary classification is used in Logistic regression. Where the bool 0 value is considered to be Southern route and bool 1 value is considered as Northern route.

Route Identification experiment setup:

- **Hyperparameter Optimization to be tested:**
    1. Defaults values of the machine learning Model (No tuning).
    2. Random Search Algorithm.
    3. Bayesian Optimization Algorithm.
    4. Tpot Genetic Algorithm.
- **Machine learning Algorithm used:**
    1. Logistic Regression.
    2. Number of iterations of testing the performance: 5.
- **Performance metrics:**
    1. Precision.
    2. Recall.
    3. F1-Score.
    4. Accuracy.
    5. Duration (Seconds).

- **Hyperparameters tuned and their 'search space':**
    1. C: Range between 0.01 to 100.
    2. Maximum iterations: Range between 10 to 2000.
    3. Solver: 'Newton-cg', 'lbfgs', 'liblinear',' sag', 'saga'.
    4. Penalty: 'None', 'l1', 'l2', 'Elasticinet'.

For tuning the Hyperparameters using Random search, Bayesian search and Tpot genetic HPO algorithms are used. The Hyperparameter tuned in Random search and Bayesian search are 'Max_iterations' for 5 different tasks. Whereas the Hyperparameters tuned in Tpot genetic algorithms is 'Population Size' for five different tests.

The optimal hyperparameters are used in order to do an experiment without having problems like memory leaking or crashing of the jupyter notebook kernel. The hyperparameters tuned for Route identification in all the 5 tasks are shown below

Test number	Max Iterations for Random and Bayesian	Population Size for Tpot algorithm
1	2000	20
2	1000	10
3	200	20
4	100	10
5	50	5

Table 5: Tuning hyperparameter for Route identification

The performance matrices in any classification algorithms are Precision, Recall, Accuracy, F-1 Score. To calculate these performance matrices, True Positive (TP), True Negative (TN), False Positive (FP), False Negative (FN) are required.

In the following, the definitions of [16] for the performance matrices are used.

- Precision: It represents "what number of selected data items are relevant". i.e. Precision is the ratio of correctly predicted positive observations to the total predicted positive observations.

$$\text{Precision} = \frac{TP}{TP+FP} \quad \text{.....(2)}$$

- Recall: It is also known as sensitivity and it is the ratio of correctly predicted positive observations to the all observations in actual class or recall equals the number of true positives divided by the sum of true positives and false negatives.

$$\text{Recall} = \frac{TP}{TP+FN} \quad \text{.....(3)}$$

- Accuracy: It is the most used performance matric in any classification ML model and it represents ratio of correctly predicted observation to the total observations.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad \text{.....(4)}$$

- F1 score: In order to calculate F1 score, it takes Precision and Recall into consideration. F1 score represents the weighted average of Precision and Recall. F1 score is also known as F1 measure.

$$\text{F1 Score} = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad \text{.....(5)}$$

5 Results

In total there are five tests/tasks done for the prediction of the ETA of the vessel. In each test some of the Hyperparameters have been tuned. For Random Search and Bayesian Search HPO algorithms 'N_estimators' have been tuned in each tasks. For Tpot HPO algorithm the tuning parameter is 'Population size'. The duration of training the dataset also depends upon the specification of the computer used.

ETA Prediction of the vessel:

The Duration of the test and the Root Mean Squared Error (RMSE) of each test with the HPO algorithms are shown below

1. HPO: none (No tuning).

Test Number	Machines Used (Specifications)	Durations (Seconds)	Root Mean Squared Error (Hours)
1	TUHH Lab	10	0.048
2	TUHH Lab	5.49	0.039
3	TUHH Lab	8.6	0.0391
4	TUHH Lab	10.58	0.0413
5	Azure Notebook (4 core, 24gb RAM)	23.69	0.038

Table 6: ETA prediction without HPO

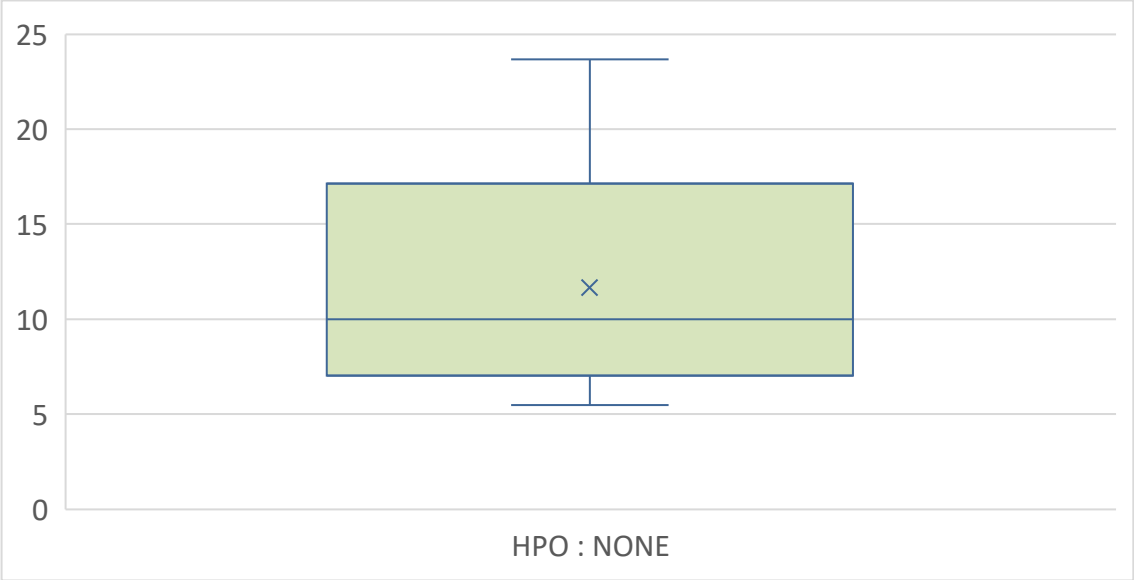

Figure 7: Training Duration without HPO

2. HPO: Random Search.

Test Number	Machines Used (Specifications)	Durations (Seconds)	Root Mean Squared Error (Hours)
1	TUHH Lab	148.09	0.053
2	TUHH Lab	392.26	0.047
3	TUHH Lab	267.97	0.046
4	TUHH Lab	650.42	0.0441
5	Azure Notebook (4 core, 24gb RAM)	334.49	0.0518

Table 7: ETA prediction with Random Search HPO

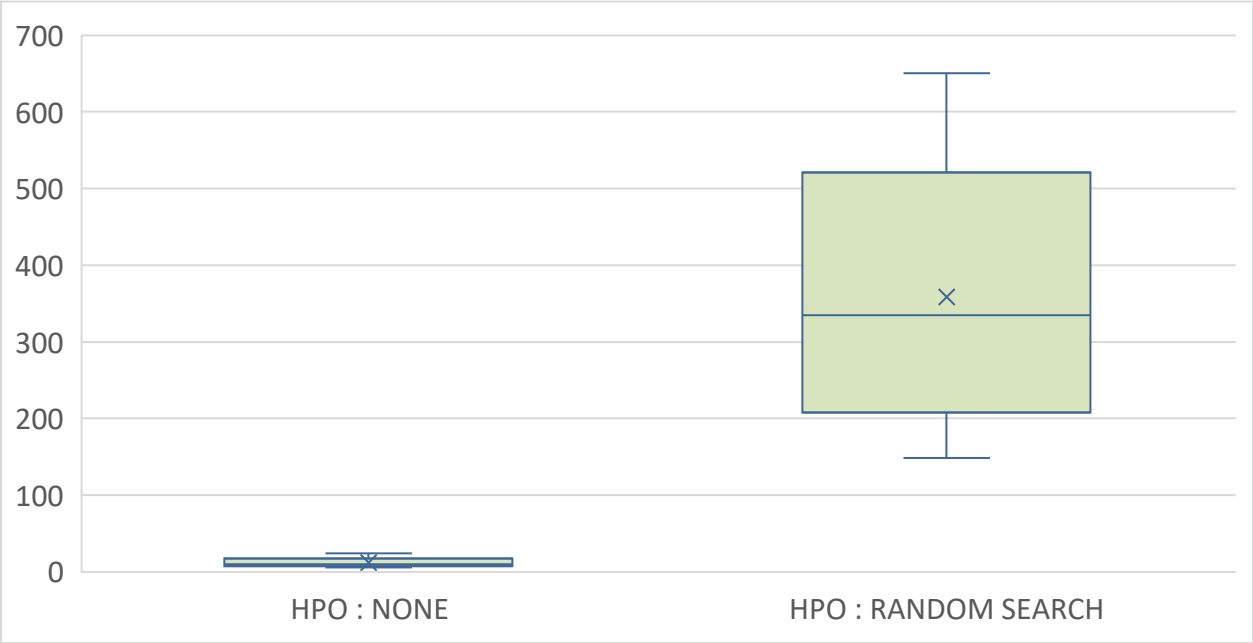

Figure 8: Training duration with Random Search HPO

3. HPO Bayesian Search

Test Number	Machines Used (Specifications)	Durations (Seconds)	Root Mean Squared Error (Hours)
1	TUHH Lab	1731.67	3.39
2	TUHH Lab	1711.33	3.35
3	TUHH Lab	1340.3	4.6
4	TUHH Lab	12442.4	4.25
5	Azure Notebook (4 core, 24gb RAM)	2703.36	5.48

Table 8: ETA prediction with Bayesian Optimization HPO

4. HPO Tpot Genetic Algorithm

Test Number	Machines Used (Specifications)	Durations (Seconds)	Root Mean Squared Error (Hours)
1	TUHH Lab	8942.02	0.024
2	TUHH Lab	2917.34	0.042
3	TUHH Lab	50400	0.0215
4	TUHH Lab	10524.84	0.011
5	Azure Notebook (4 core, 24gb RAM)	40682.56	0.0155

Table 9: ETA prediction with Tpot Genetic Algorithm HPO

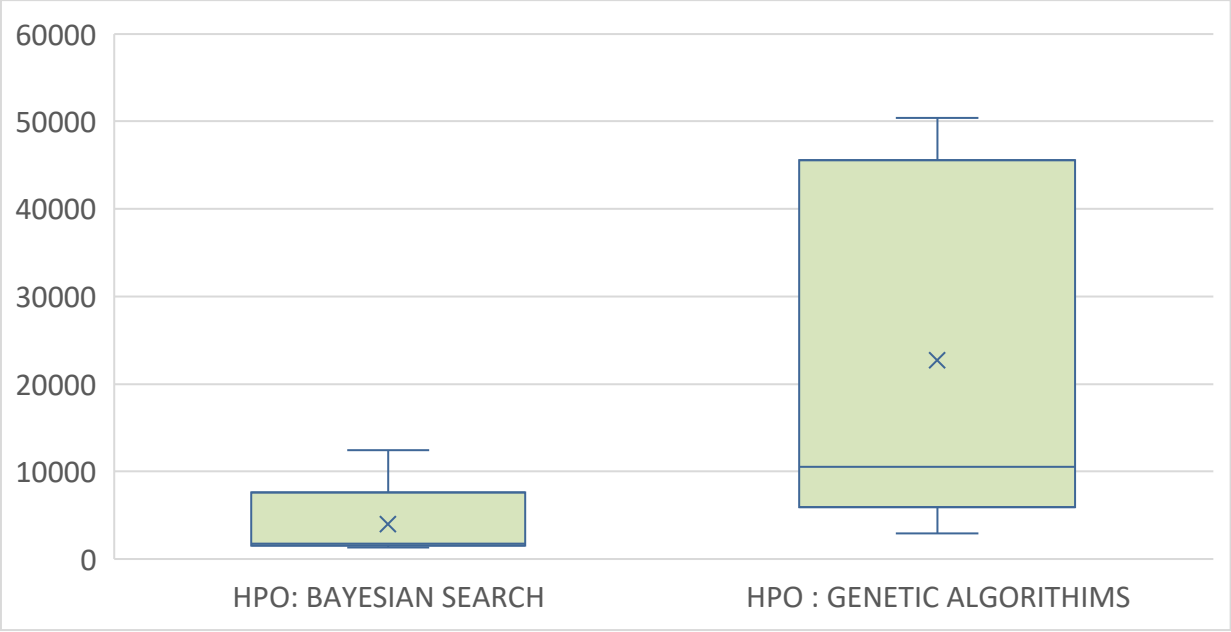

Figure 9: Training duration with Bayesian and Tpot Genetic Algorithm HPO

RMSE of the tests can be seen below

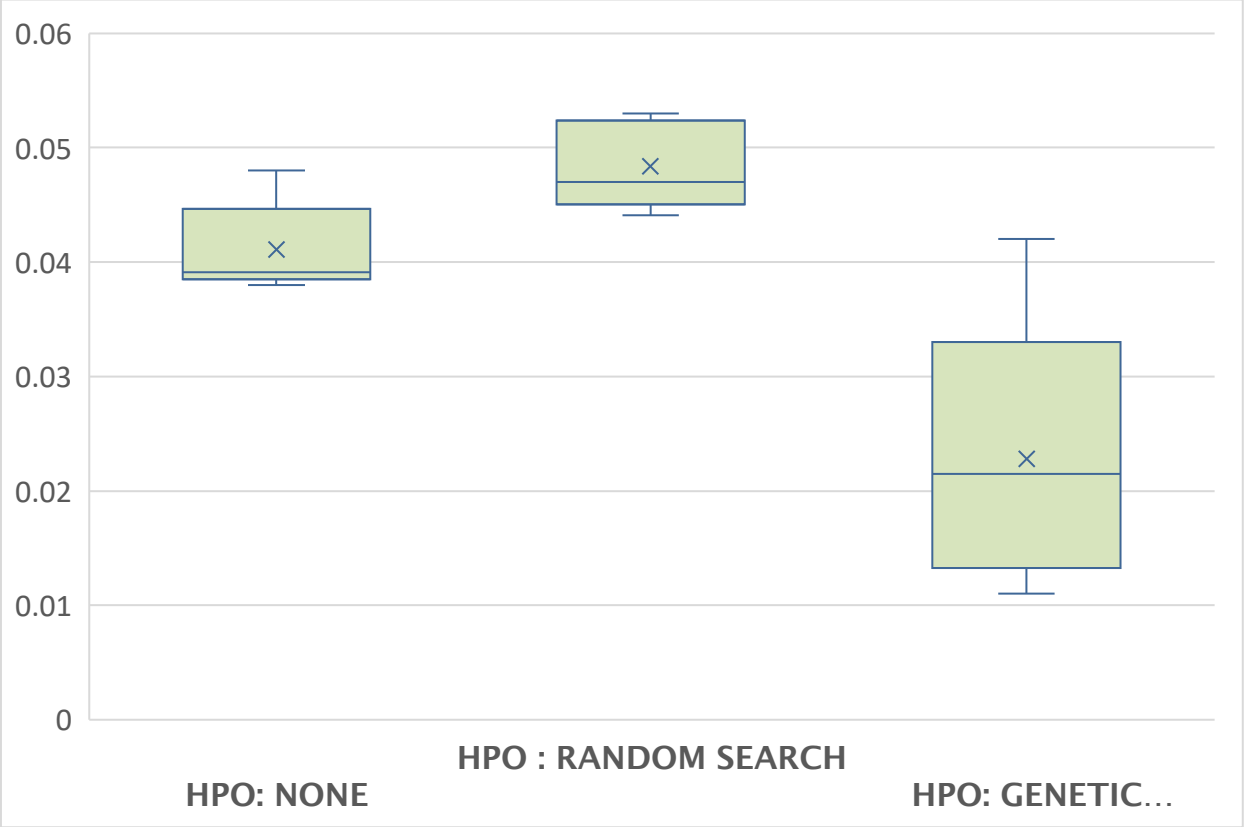

Figure 10: Comparison of RMSE for Different HPO algorithms

Route Identification of the Vessel:

The tuning parameters for Route Identification is 'Max_iterations' for Random and Bayesian Search. Whereas the tuning parameter for Tpot Genetic Algorithm is 'Population size'.

The performance matrices for the route identification are:

- Precision.
- Recall.
- Accuracy.
- F1- Score.
- Duration.

These Hyperparameters have been tuned for five different tests and the results are shown below:

1. HPO none (No tuning)

Test Number	Machines Used (Specifications)	Precision	Recall	Accuracy	F1-Score	Duration (Seconds)
1	Azure Notebook (4 Core, 24GB RAM)	0.73	0.56	0.79	0.55	0.56
2	TUHH Lab	0.73	0.56	0.79	0.55	0.52
3	Google Colab*(2 Core, 12 GB RAM)	0.73	0.56	0.79	0.55	1.14
4	Google Colab*(2 Core, 12 GB RAM)	0.73	0.56	0.79	0.55	1.14
5	Google Colab*(2 Core, 12 GB RAM)	0.73	0.56	0.79	0.55	1.44

Table 10: Route Identification without HPO.

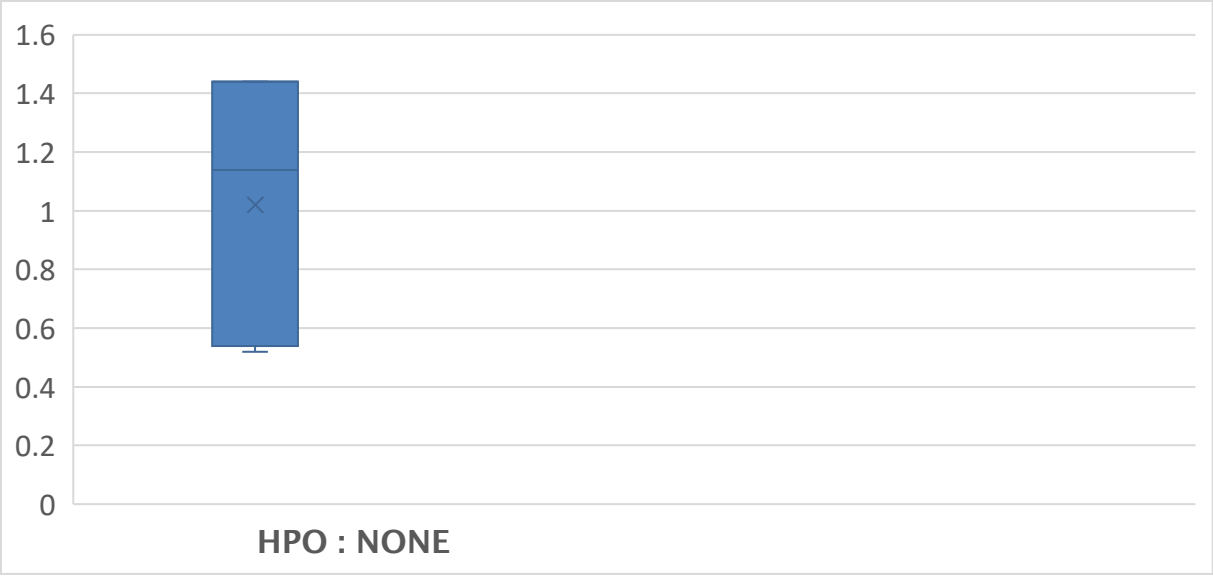

Figure 11: Training duration without HPO

2. HPO Random Search

Test Number	Machines Used (Specifications)	Precision	Recall	Accuracy	F1-Score	Duration (Seconds)
1	Azure Notebook (4 Core, 24GB RAM)	0.78	0.66	0.82	0.69	43537.44
2	TUHH Lab	0.78	0.66	0.82	0.69	2246.23
3	Google Colab*(2 Core, 12 GB RAM)	0.79	0.65	0.82	0.67	173.44
4	Google Colab*(2 Core, 12 GB RAM)	0.78	0.66	0.82	0.69	449.47
5	Google Colab*(2 Core, 12 GB RAM)	0.78	0.66	0.82	0.69	1036.90

Table 11: Route Identification with Random Search.

3. HPO: Tpot Genetic Algorithm

Test Number	Machines Used (Specifications)	Precision	Recall	Accuracy	F1-Score	Duration (Seconds)
1	Azure Notebook (4 Core, 24GB RAM)	0.98	0.97	0.98	0.98	4058.01
2	TUHH Lab	0.98	0.97	0.98	0.97	1806.63
3	Google Colab*(2 Core, 12 GB RAM)	0.99	0.97	0.98	0.98	1653.37
4	Google Colab*(2 Core, 12 GB RAM)	0.98	0.98	0.98	0.98	1902.58
5	Google Colab*(2 Core, 12 GB RAM)	0.97	0.99	0.99	0.98	6583.36

Table 12: Route Identification with Tpot genetic algorithm

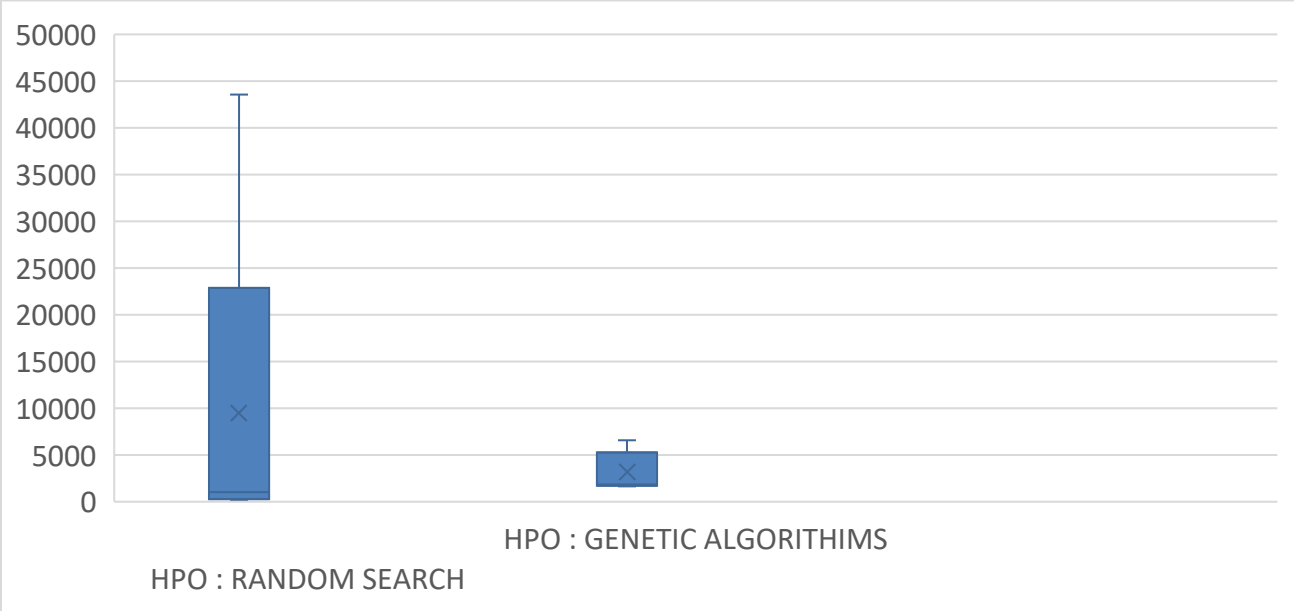

Figure 12: Training duration for Bayesian and Tpot Genetic Algorithm

4. HPO: Bayesian Search

Test Number	Machines Used (Specifications)	Precision	Recall	Accuracy	F1-Score	Duration (Seconds)
1	Azure Notebook (4 Core, 24GB RAM)	0.73	0.56	0.79	0.55	112.39
2	TUHH Lab	0.73	0.56	0.79	0.55	117.55
3	Google Colab*(2 Core, 12 GB RAM)	0.39	0.50	0.77	0.44	96.52
4	Google Colab*(2 Core, 12 GB RAM)	0.39	0.50	0.77	0.44	118.51
5	Google Colab*(2 Core, 12 GB RAM)	0.30	0.50	0.77	0.44	215.62

Table 13: Route Identification with Bayesian Optimization.

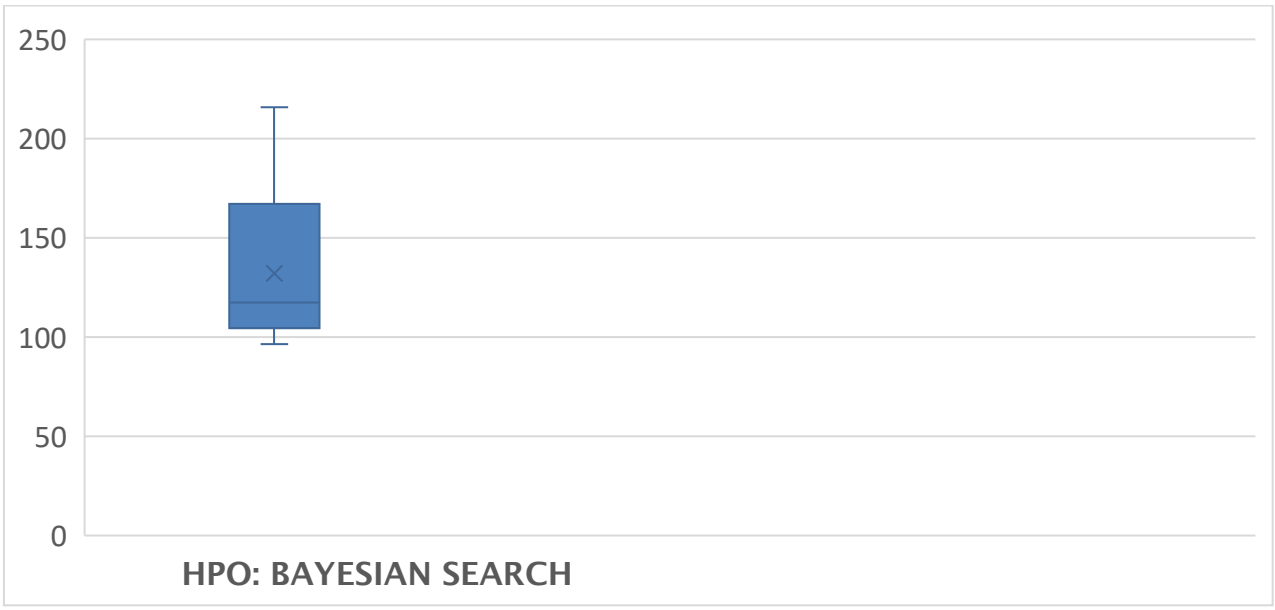

Figure 13: Training duration with Bayesian Optimization.

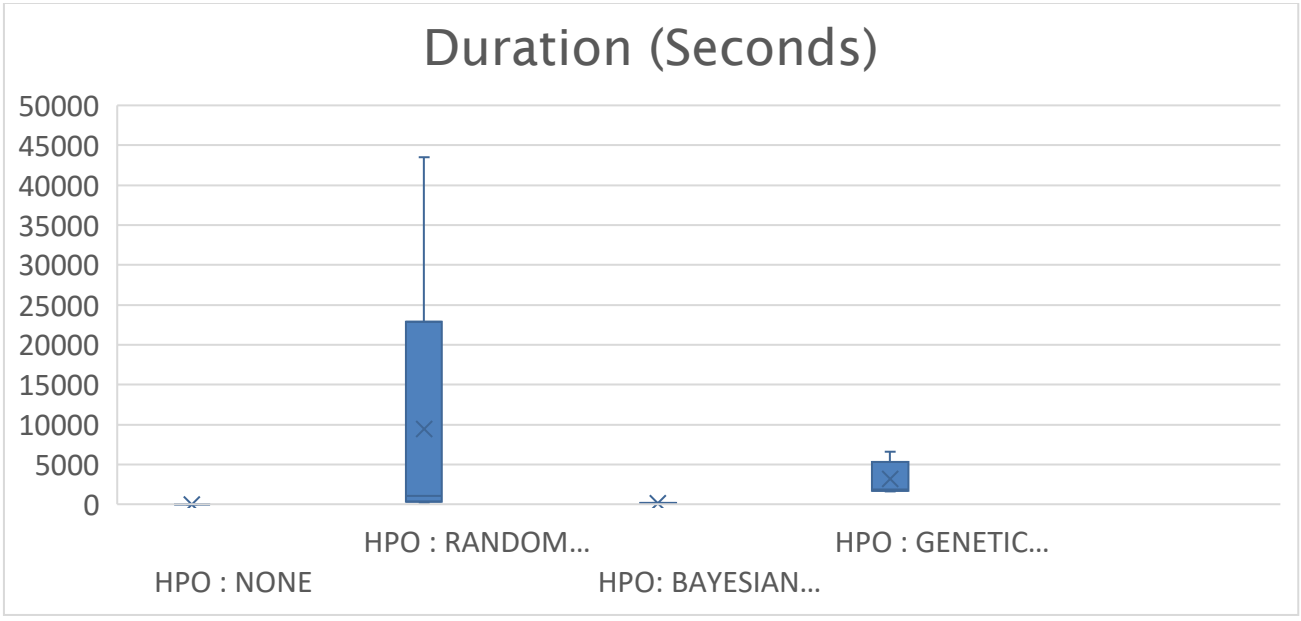

Figure 14: Comparing different HPO algorithms duration

The comparison of different HPO algorithms results is shown below

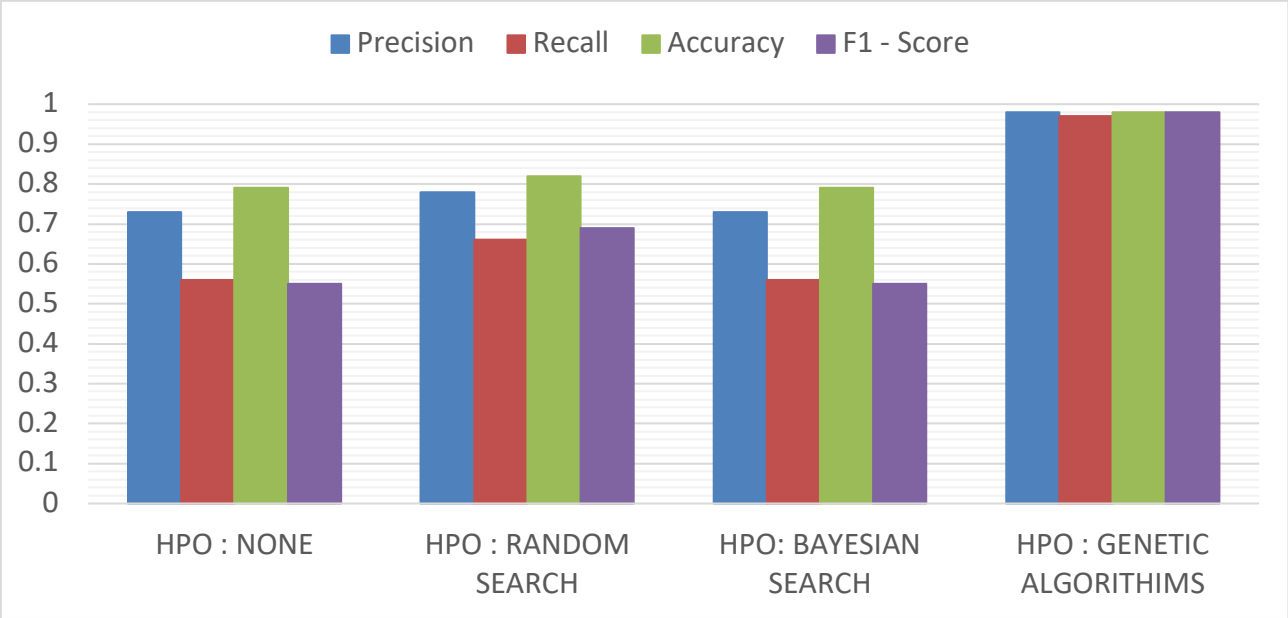

Figure 15: Performance matrices of different HPO

After many attempts of training, there are certain points which are very important worth mentioning-

1. Pre-processing is a vital part of machine learning.
2. Visualization of the Data is essential.

The training of the dataset is done using three different types of machines. They are Azure Notebook, University Jupyter Notebook and Google Notebook. Based on the machine capability, different set of parameter combinations is performed on five different tasks for each objective. The training of the dataset is done with the help of regression models. Figure 8 and 13 represents the training duration of the dataset with HPO algorithms. From Figure 8, it can be seen that “Tpot genetic algorithm” HPO has taken the highest training duration of the dataset for prediction of ETA of the vessel. From Figure 13, it can be observed that Random Search HPO algorithm has the maximum training duration to predict the Route of the vessel. Each of the two objective has different performance metrics. Performance metrics are used to compare the effect of HPO algorithms on the regression model.

5.1 Conclusion

For Route Identification of the vessel, the primary performance metrics is ‘Accuracy’. As mentioned in the results, with logistic regression training model, the best accuracy is observed using “Tpot genetic algorithm” HPO optimization. The accuracy observed in the model is 99%. With no tuning the observed accuracy is 79%. With HPO, the improvement is almost 20% higher than the accuracy without HPO. By tuning the ‘generations’ and ‘population size’ parameter, the result has been achieved. It can be said that, the higher the parameter, the better the results. Besides uninterrupted higher computing power is also essential part of this process. Memory leaking can occur if the tuning parameter crosses the limits of the computational power. During the experiments, after training for days, jupyter lab crashed due to this situation.

For prediction of ETA of the vessel, Random Forest regressor is used to train the model. The ETA is measured in hours. Root mean squared error is used as performance metrics. Lower root mean squared error means better results. In the ETA training one feature played a vital role. The feature name is ‘speed’. Without this feature the overall training results were not up to the mark. It is also observed that, Tpot genetic HPO algorithm has better results compared to other HPO algorithm used in the experiments. With five consecutive tasks proved that tuning the parameter of HPO has an effect of the overall experiment’s setup. For genetic algorithm, the parameters tuned are “Generations” and “Population Size”. The last experiment using Tpot genetic algorithm HPO has the lowest root mean squared error which is around 0.0155 in hours. With no tuning the root mean squared error is around 0.038 in hours. The improvement is almost 59.21% percent which is very significant.

From the observation in both of our objective experiments, we can eventually say that HPO has a great effect on improving the machine learning model’s performance matrices. The genetic HPO algorithm has proved to be one of the best options to predict ETA of the vessel as well as the Route Identification of the vessel.

Future Research

The future improvements can be done using many other performance metrics available to test ETA training and trying different regression models. Due to time constraints and machine constraints other performance matrices were not used in this project. With better machine learning server, the results can be widely improved in future.

6 References

1. Harati-Mokhtari A, Wall A, Brooks P et al. (2007) Automatic Identification System (AIS): Data Reliability and Human Error Implications. J Navigation 60: 373–389. https://doi.org/10.1017/S0373463307004298
2. Hutter F, Kotthoff L, Vanschoren J (2019) Automated Machine Learning. Springer International Publishing, Cham
3. Probst P Hyperparameters, Tuning and Meta-Learning for Random Forest and Other Machine Learning Algorithms
4. James S. Bergstra, Rémi Bardenet, Yoshua Bengio et al. (2011) Algorithms for Hyper-Parameter Optimization. In: pp 2546–2554
5. Booth A, Sutton A, Papaioannou D (2016) Systematic approaches to a successful literature review, Second edition. Sage, Los Angeles
6. Frank Hutter, Holger H. Hoos, Kevin Leyton-Brown (2011) Sequential Model-Based Optimization for General Algorithm Configuration. In: Springer, Berlin, Heidelberg, pp 507–523
7. Rasmussen CE, Williams CKI (2005) Gaussian Processes for Machine Learning. The MIT Press
8. James Bergstra, Yoshua Bengio bergstra12a.dvi
9. Parmentier L, Nicol O, Jourdan L et al. TPOT-SH: A Faster Optimization Algorithm to Solve the AutoML Problem on Large Datasets: 471–478. https://doi.org/10.1109/ICTAI.2019.00072
10. Fernández-Delgado, M., Cernadas, E., Barro, S., & Amorim, D. (2014) Do we Need Hundreds of Classifiers to Solve Real World
11. Probst P, Bischl B, Boulesteix A-L Tunability: Importance of Hyperparameters of Machine Learning Algorithms
12. Abbas Harati-Mokhtari, Alan Wall, Philip Brooks et al. (2008) Automatic Identification System (AIS): A Human Factors Approach
13. Pietrzykowski Z, Wolejsza P, Magaj J (2015) Navigators' Behavior in Traffic Separation Schemes. TransNav 9: 121–126. https://doi.org/10.12716/1001.09.01.15
14. Jasper Snoek, Hugo Larochelle, Ryan P. Adams Practical Bayesian Optimization of Machine Learning Algorithms

15. Simon, Author Evolutionary Optimization Algorithms
16. Vakili M, Ghamsari M, Rezaei M (2020) Performance Analysis and Comparison of Machine and Deep Learning Algorithms for IoT Data Classification