UiT The Arctic University of Norway

Department of Industrial Engineering

Machine Learning in Predictive Quality Management

Simbarashe Keith Chikwekwe

Master's thesis in Industrial Engineering - INE 3900 – May 2024

Abstract

This thesis investigates how machine learning techniques has greater advantages over traditional Statistical Process Control (SPC) methods and Anomaly Detection (AD) in monitoring and controlling process parameters so that the process remains stable and in control to meet required quality satisfactions. The broader term of Predictive Quality Management introduces an important subject of predictive analytics which can be collaborated with machine learning tools to predict or forecast future process patterns. This thesis employs a case study, of fishing industries that amass large marine data in form of Automatic Identification Systems data, catch data and environmental but fail to draw meaningful correlations between data variables towards sustainable and efficient use of fishing vessel resources. Different machine learning models were implemented, techniques for data cleaning and preprocessing provided a leeway to draw patterns and trends in our dataset. and a performance evaluation using suitable metrics was conducted to determine which regressor algorithm predicts and generate forecasts for location of fish species. This thesis contributes to a deep understanding of data analysis and offers recommendations to decision makers.

Keywords: statistical process control, predictive analytics, anomaly detection, random forest, machine learning, metrics, algorithm.

Table of Contents

List of Tables

List of Figures

Acknowledgements

Firstly, I would like to express my profound gratitude to my supervisor Espen H. Johannessen, for his unwavering dedication and support throughout the thesis period. His immense contribution and effort in providing guidance and direction can never go unmentioned.

Secondly, I would like to honor my guardian, my mentor, my grandfather and my English teacher John W. Neganje, who was promoted to higher glory in February 2024. May his dear departed soul rest in eternal peace. It was a difficult period for me to travel to my home country, Zimbabwe in a short space of time for the funeral and l had to put a halt to my thesis progression. I dedicate this master's thesis to him with all due respect.

I extent my sincere thanks to the companies for their collaboration in providing case studies, industrial knowledge and ensuring data is available for this research to be successful.

To my friends and family, you have been a pillar of strength through providing love and encouragement throughout this demanding exercise.

Finally, I acknowledge the works that professionals and researchers do in providing scholarly material that is useful, up to date and enriched with current technologies.

1 Introduction

The continuous confrontation of manufacturing companies by new technological possibilities, globalisation, and increased competition in the era of Industry 5.0, has left them to adapt to modern business trends in pursuit of customer satisfaction Traditional manufacturing practices and production landscapes are then subject to redefinition. In this context, Quality 5.0 seeks to supersede the shortcomings met by using conventional quality control methods [1]. Some explanations are needed to know how businesses can preserve or improve the quality of their products and services as well as simultaneously accelerating efficiency and reduction of costs hence an introduction of machine learning in quality management is a necessary approach towards quality control and assurance[1, 2]. Artificial intelligence through use of machine learning is also taking a centre stage in quality assurance issues [3]. The application of machine learning, deriving form process data has therefore helped firms in making data driven estimations/ decisions on product quality. A key point for predictive quality management is to control process stability, which means that the process is normal and in control. From this process behaviour the control limits are calculated, and one can then see the specification limits relative to the control limits. This outputs the capability of the process. The idea of machine learning in predictive quality management now involves amassing of information in form of sensory data (analogue sensors or multivariate data), process parameters, product quality and design factors etc from a manufacturing process and then train the model in order to come up with model deployment and predictive algorithms for control charts that will result better decision making [1, 4].

1.1 Background

Early procedures of quality focused on visual inspection of parts or products after manufacturing. The greatest disadvantage with this approach was that faults were found postproduction and there are extra costs associated with reworks. This was further developed into Total Quality Management (TQM) an approach to management mainly focusing on systematic process management, customer-centered, continuous improvement, partnering with suppliers collaboration [5]. Companies that adopt Quality Management gather information that assists them in identifying weaknesses, strengths, possible areas of improvement and faults. The major goals are to determine a high quality, high performing products and services that meets and exceeds internal and customer expectations. Benefits of a well-integrated Quality Management System are product performance, reduced wastes, customer satisfaction, enhanced productivity and increased revenues [6].

Major challenges that have arisen in QM, are consequences related to managements competence, readiness for business risk, competitive orientation, adaptability to business changes after collecting large amounts of process data or factors but not knowing how to make able decisions for future trends. This has led to most business in increased costs and less profitability [7].

In TQM, there exists a robust and flexible combination of statistical tools to monitor, control and stabilize a process by minimizing variability among the product called Statistical Process Control (SPC). Hence, this method is applied by examining a sample size n at different intervals t and the current produced result k compared against required standards. The use of control limits (upper and lower) is regarded as the process voice for a proactive approach. Additionally, a control chart is drawn where process observations are presented on a graph with limits to determine unwarranted behaviors and trends. The voice of the process is centered on behavior relatively to the control limits. For a process to be ruled in or out of control, a probability is determined based on the distribution of variable measurements. Examining the posterior process requires that for each inspection of samples we need to determine the process limits to ascertain whether it is **out** or **in** control process. This renders Statistical Process Control as a reactive approach rather than preventive [8, 9].

Shift towards Predictive Quality Management (PQM) has been necessitated by the desire to have a proactive quality control approach (Quality 5.0) that anchors on main pillars of predictive analytics, real time monitored processes, AI driven process optimization and collaboration between human and machines. Quality control is embedded at each stage in the process and defects are eliminated before they occur. Benefits of this approach to quality include a clear understanding of the production process, collaborative nature of decision making, maximized product quality and an increased efficiency. Manufacturing companies generate big data of their processes which results in large data sets but fail to use analytical tools in order to come up with correlation of the data in order to make valuable decisions, hence the emergence of Machine Learning and Artificial Intelligence proposed a great shift by ensuring real time identification and correction of defects, reduction in downtime, improved accurate inspections, minimal wastes and improved process reliability. Data analytical techniques together with machine learning can be employed to analyze given big data to single out trends and identify shift in patterns which points out to inherent quality challenges therefore paving way to proactive quality control [10].

A good example is an AI predictive quality solution by Craftswork Company as mentioned on their website, that aimed at reducing scrap and increasing efficiency for a brick production company. The client needed a reduction in scrap rate in the production plant. The facility produced scrap material or faulty bricks and it was difficult to link to its root causes. The major challenge was that quality control was conducted at end of the production process making it unpredictable and inefficient.

Craftwork were able to identify patterns in the data by merging quality data, machine and sensor data from different production steps. Adding on that, their AI predicts possible issues in manufacturing beforehand. The advantage of using AI driven technology ensures real time data insights and a significant reduction rate in failures. Production of high-quality bricks then add as a competitive advantage leading to increased market share and customer trust.

1.1.1 Case: Fishing Industries

Fishing in the sea is a complex process which is multivariable dependent. A lot of fishing companies have found the means to gather data (data mining) in an effort of predicting the movement patterns of fish, when likely to produce big yields, favorable weather conditions and even type of fish. The Norwegian Sea is a particular example of a fishing platform where vessels operate daily. The data that fishing companies collect requires aid of machine learning to predict behavior of fish according to variables like weather, time of day, windspeed and direction to name a few. The prediction will enable fishing companies to detect anomalies in behavior and make informed decisions before vessels set out. Given the availability of data concerning fish behavior, it is prudent that an algorithm that can use such data to predict availability and location of fish be developed to maximize each catch yields for fishing companies.

From this pool of candidate sites, a smaller sample group is taken by using formerly surveyed sites that are accessible and previous catches recorded. Climate and weather data, stream direction, size of catchment area and type of land are some of the environmental factors used for each candidate site using GIS. This enables validation of selected sites using maps and creating individual maps to show an actual geographical representation of the sites. This helps companies in making evidence-based decisions in knowing were to set out the vessels for a fishing exercise [11].

1.2 Problem Statement

Oddvar Nes is a coastal fishing company located in Botnhamn on Senja, Norway. They own and operate coastal spinners “Trondskjær” and “Lise Beate”. They face difficulties in making use of big data obtained for their fishing fleet which is of great importance in determining fishing forecasts in terms location and when fishing is mostly likely to take place. The big data in a fishing environment includes possible relevant data sources from AIS tracking, departure notices, catch notifications, registrations, banknotes, time, weather, currents, temperature, moon phase, bait (in the sea from capture samples), sonar data among other activities (seismic, defense exercises etc.). This has led them to experiencing time and fuel losses hence poor yields or no catches resulting in high operating costs. The poor yields result from lack of comprehensive use of mined data to predict location and behavior of fish.

The successful development of an algorithm that can predict the actual behavior of fish from a spectrum of different variables will be of great significance to fishing companies which would be able to direct their fishing vessels to locations where they can experience high catches.

This thesis aims to explore how big data can pave way for streamlined capture processes that will contribute to increased sustainability, cost optimization and efficient lean based solution in fishing vessels. his thesis will employ an anomaly detection machine learning prediction model to predict location of fish on Norwegian coasts according to wind speed, wind direction, time of year etc. and all necessary big data included.

1.3 Project benefits

The benefits of this project are to a greater extent aligned to an improvement of the rate of production in fishing industry, but significant benefits lie in use of machine learning algorithms whose advantages are:

1. Increased sustainable and efficient fishing processes. Being able to predict accurately the location of catch and time saves unproductive operational costs (fuel, time, labour).
2. Predictive maintenance can be employed to fishing vessels. The days with poor catches, the spinners are laid idle for servicing and routine maintenance.
3. Being able to predict yield can encourage companies to make data driven decision making for example innovative fishing practices, risk mitigation on safety hazards and failure rate of equipment and fuel consumption is reduced.
4. Minimal environmental implications by fishing operations

5. Reduced waste and making sure the yield meet the quality standards and customer specifications.

1.4 Hypothesis

- Machine learning based prediction of fish behaviour will direct fishing companies to higher yields.
- Predictive quality management in fishing practices will adhere to regulatory framework including government policies, regulated fishing zones and legal requirements.
- Guaranteed reduction of wastes leading to high catch quality.

1.5 Assumption

This study assumes that models are specific according to fish species provided in the dataset. It also assumes that the training data set contains all variables and prediction is constant for all similar conditions.

1.6 Objectives

a. To model, train and test a predictive algorithm which can analyse multiple variables and predict location or abundance of fish in sea.
b. To verify the algorithm modelled in (a) above.
c. To establish relationships between static and predictor variables in our dataset.
d. To select the best predictive algorithm(s) based on performance metrics.

1.7 Scope

The student will conduct a survey on the use of SPC in manufacturing industries to establish a link to the case study area and find the existence of a research gap. Additionally, a survey on data mining in fishing companies in Norwegian sea to apply SPC to detect anomalies.

1.7.1 A review on the use of data in SPC and machine learning

Particular attention will be given to the review of anomaly detection in manufacturing considering its relation machine learning and SPC. The student will review information methods, types. In this context of machine learning, data usage implies training of data sets to create an algorithm which is either supervised or unsupervised. The student will also focus on

feature selection where different methods are used to define and select the best predictor variables.

1.7.2 Data availability

The student will also carry out surveys and interviews to fishing companies to establish patterns of yield without prediction tools to justify the need for machine learning prediction tool, and to reveal how the data available is used by the fishing companies today. Student will commence on experimental set ups to train data sets from fishing companies to link with ongoing review of literature.

1.7.3 References/Links

The following links gives a good start point on literature review of statistical process control and anomaly detection in manufacturing and good point of establishing a research gap. Other websites provide fishing data variables in Norwegian Sea in terms of locations, catch data, landing data only to mention a few.

1. https://se.mathworks.com/discovery/anomaly-detection.html
2. https://acerta.ai/blog/anomaly-detection-in-manufacturing/
3. Front page (barentswatch.no)
4. https://kystdatahuset.no
5. Transparency for a Sustainable Ocean | Global Fishing Watch

1.8 Organization

This stage articulates main activities and the deliverables to be carried out. Project monitoring is done in conjunction with supervision from the project coordinator per each main tasks completed and presented through status reporting in form of progress reports. Consultation with external stakeholders will be done as part of data collection.

Plan	Milestones	Main Activities & Deliverables
Part 1	16.10.23	Official start date of thesis
	08.11.23	Task Description 1 • Introduction • Scope • References
		Pre-study report
		Literature review • Statistical process control (SPC) • Anomaly detection in manufacturing (ADM) • Data collection (surveys and interviews) • Link between SPC and ADM • Supervision of a process control system by machine learning
	08.12.23	Progress report
	11.01.24 01.2024	Delivery of Thesis Part one • One status report • Presentation of Thesis Part one
Part 2	15.01.24 09.02.24 04.2024 15.05.24	Official start date of Thesis part two Task description two (Introduction, scope, references) Pre-study report two Methodology • Data collection • IDE implementation (Training, analysis, and validation) • Performance evaluation • Discussion of results • Conclusions and Future work Progress Report (Delivery and presentation) Documentation of study and final presentation

Table 1: Distribution of thesis progress

2 Literature Review

This chapter provides a broad-based review information from scholarly articles pertaining to data mining, acquisition, processing and application in fisheries around the world towards an efficient and sustainable fishing industry. The main purpose of this section is to establish a research gap whereby it is made clear that the existing literature has a void relating to Statistical Process Control`s role in detecting anomalies in a system which can be known before application of machine learning method(s) to predict and track processes in fishery industries, in our case particularly two vessels operating in the Norwegian sea.

2.1 Data Analytics in Fisheries

A key component of agenda of 2030, particularly some specific Sustainable Development Goals (SDGs) one and two, only to mention a few, focuses on the sustainability of biotic resources, such as fish, according to 2021 International Resource Panel Report. According to a Vietnamese study by Nguyen and Tran (2019), migrating species like oceanic tunas are protected by management system that is well integrated and it supports the management of multi species ecosystems and their optimization of multi equipment fishing. This bolsters an abundance of research that shows sustainable fisheries management to be an all-encompassing preserving and balancing fish populations, lowering pollution to ecosystems and combating unregulated practices.

Research has indicated that in order to buck the trend, global sustainability guidelines must be integrated into the operations of all players, particularly the major fishing businesses. In an attempt to catch enough fish yield to satisfy the soaring demand, large fishing companies and dealers are putting freshwater and marine habitats under unprecedented strain. Furthermore, there are still several problems that make it difficult to manage sustainable fishing and encourage ocean stewardship, including lack of accurate and useful information and lack of funding by many governments to process and analyze data on types, locations and times that fishing boats are catching in addition to other human activities at sea. Because the targeted species are currently in short supply, it takes five times as much work (measured in kilowatt hours) to catch the same quantity of fish as it used to happen in early ages. This shortfall seriously imperils both the financial sustainability of fishing and the ability of endangered oceanic species to procreate and sustain their populations. Furthermore, putting in place efficient procedures and frameworks for the management of fisheries and the dearth of information necessary to measure catches and identify the most effective fishing methods,

particularly in developing nations. Practices like cutting back on the fish fleet, for instance in Asia, haven't resulted in more sustainable outcomes rather multiplied IUU fishing activities.

2.1.1 Machine Learning in fisheries

Artificial intelligence augmented with automation strategies has played a significant role in monitoring marine resources. ML algorithms can be integrated with underwater imagery systems to identify and classify marine animals [12]. Noise data signals can be utilized in tracking their movement [13]. These techniques can raise the effectiveness and accuracy of monitoring systems. These methods can also be used to get additional knowledge about the behavior and ecology of marine life. To foster the development of marine protected zones and manage fisheries a number of AI-based support solutions are developing [14]. These systems can help ensure that maritime resources are managed sustainably to safeguard diversity of aquatic life. AI and automation are also having impact on marine science in terms of forecasting and predicting the state of the ocean, weather, tides and ocean currents. ML algorithms are being adopted to predict patterns with a high degree of accuracy. By utilizing these, maritime operations can be made safer.

Through use of automation, data is being collected and analyzed in the marine fraternity. In water bodies, in zones that are unreachable data variables are gather through use of drone and autonomous underwater technologies hence making it easy to collect variables in spatially pronounced places [15].

Deep learning techniques like YOLO can be used to monitor undersea life inclusive of plants . [16].

2.2 Empirical Studies Review

2.2.1 Statistical Process Control (SPC)

The concept of SPC was introduced by Dr. Walter Shewhart of Bell Laboratories in 1920 and later introduced into Japanese Industry by Dr Deming. The usage of statistical techniques to monitor and regulate a process to make sure it runs as efficiently as possible to generate a conforming product is known as Statistical Process Control (SPC). Therefore, SPC tools are implemented for the purpose of controlling and improve of process. A process is monitored under SPC to ensure that it operates predictably and produces the greatest amount of conforming/better quality product with reduced variability and waste. Important design experiments, continuous improvement and implementation of control charts are used in SPC.

Changes in the procedure that could impact the products quality can be identified and fixed, minimizing waste and the possibility that issues will be transferred to the client. An alert is set off as a method of signalizing a change of an undefined process output, when a process is said to be out of control, prompting engineers to search and attempt to eliminate assignable reasons of variation. Proactively preventing out of control situations from occurring is more successful, allowing for process adjustments in a preventive measure to reduce the number of non-conforming items generated [17, 18].

Another definition of product quality can be regarded as meeting customer expectations and requirements and even exceedingly more, hence the need for continuous improvement and not dwell on past performance measure [19]. The deployment of SPC strategies by companies is a major anchor technique in Total Quality Management (TQM) by application of statistical methods for managing, controlling and monitoring manufacturing processes [20].

2.2.2 Successful application of SPC

Several prerequisites are important in the implementation of a successful SPC. It has been discovered that most organizations spent little time on management and implementation aspects of SPC. Instead, they solely focus on control chart matters as their main approach in handling SPC issues. Hence not much documentation has been made in companies showing a step-to-step practical guidance to foster a viable practical implementation of SPC as part of overall Quality Management System [21]. Figure 1 below is a prerequisite framework important for a successful implementation of SPC.

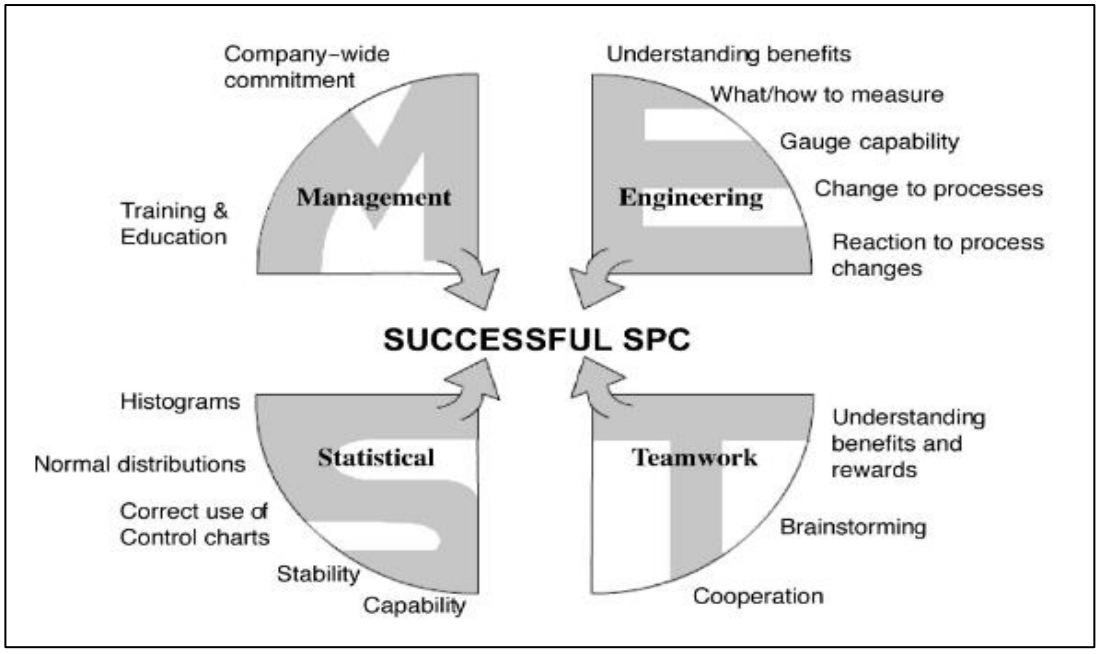


Figure 1:Prerequisite implementation of SPC [22]

2.2.3 Implementation of SPC

In SPC application, an understanding of key product characteristics which are responsible for causing process variations. The stages for implementing SPC are shown in Fig 2 below.

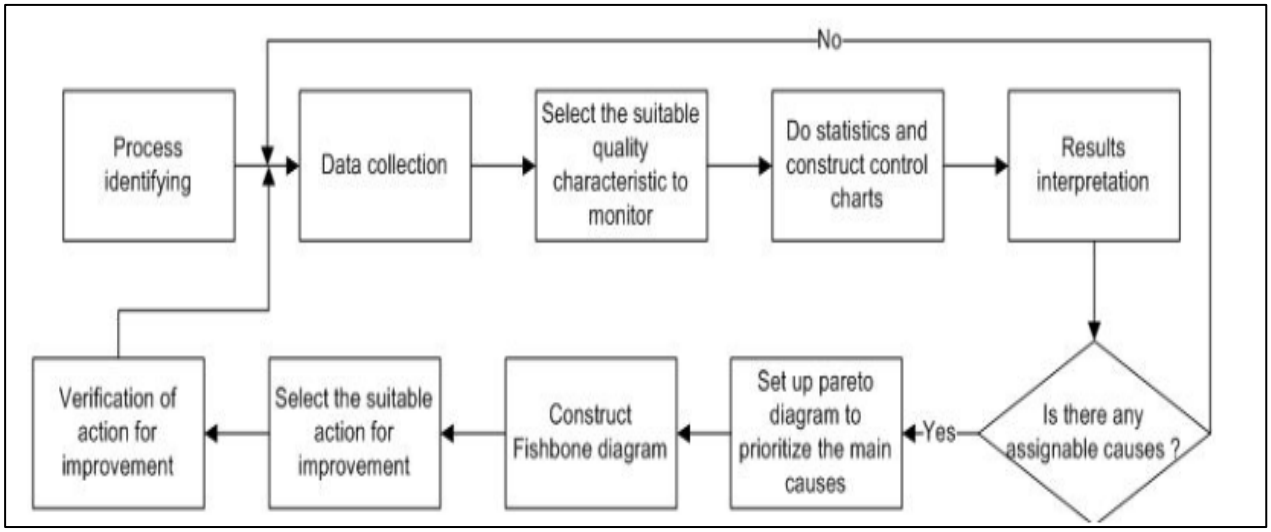


Figure 2: Stages for SPC Implementation [23]

This research provides information on how to identify assignable causes of variation[24], however, this research asserts the need to take an overall SPC implementation as problem-solving process therefore an analytical approach is required from the onset identification of assignable causes[25].

The above proposed framework which can be related to a DMAIC as a Lean Six Sigma way is further described below [26].

Process Identification: This is a preparation stage. This critical role is for managers in educating/training all levels of main goals of SPC. Companywide commitment is expected, encouragement and follow-ups from leadership in delegation of roles and responsibilities. Workers are taught to interpret control charts correctly, knowledge of statistical principles and general architecture for the problem-solving procedure. Selection and design of control chart is performed.

1. **Data Collection:** This is a data finding and recording stage of process through brainstorming, questionnaires and interviews to observe the aforementioned process to identify the quality problems. Process is SPC prioritized. A Cost Benefit Analysis is carried out. Mistake proofing techniques are also applied.

2. **Control Chart Formulation:** Statistical computations and recording methods. Maintaining of procedural computation and recording. A possible comeback for out-of-control plan.
3. **Result Interpretation:** The training stage is expected to manifest in terms reading of control charts correctly. Clear pursuit of specific obligations and competencies. High level of responsibility of workforce is expected.
4. **Assignable Causes:** Identification of data type (measurable or attributes) in terms of conforming units and non-conformities. Experience, motivation, deep knowledge and communication attributes are required to be able to identify stimulators of process variability. For a process in control, but not capable there is need to eliminate common causes of variation through Design of Experiments or Taguchi. Quality tools like fishbone diagrams and pareto analysis are employed for root cause identification. Availability of a vibrant problem-solving and speed is needed to identify stable and unstable causes in control charts.
5. **Appropriate Action for Improvement:** A data bank of possible causes and approved actions, balancing costs in terms of chosen actions and measure of efficacy of accepted actions. Management still expected to offer encouragement to responsible workers. For a process in control and capable financial terms benefits are assessed for corrective actions.
6. **Verification of Actions for Improvement:** Is a problem-solving stage of accepted improvement actions through monitoring and evaluation. Availability of gathered process information is the necessary fuel for future decision-making processes. A clear evaluation mechanism of improvement actions impact. The efficiency of such corrective actions is calculated as a percentage of realized actions that have contributed to reducing process variability.

2.2.4 Traditional methods of SPC

In order to monitor and control process variability to increase firms' competitiveness and product quality, a statistical quality control approach method is used by employing different tools and techniques. The tools can be classified in the following categories:

1. **Identifying Tools –** flowcharts, check sheet.
2. **Analytical Tools-** scatter plots, fishbone diagrams
3. **Prioritizing Tools-** pareto analysis, histogram

2.2.4.1 Control Charts

Control charts are responsible for monitoring quality characteristics/ regulated values that are related to manufacturing processes. Random variables can be separated from a systematic cause of variations. When we suspect a process needs to be investigated, to ascertain corrective actions to secure products quality in required conformance standard, control charts are employed to reduce variability of the overall process. A process implied "**statistically in-control**" produces product within reasonable variability due to random/ common causes which are of permanent dominance in our process hence contribute to overall variability. A process implied "**statistically out-of-control**" exhibits special/assignable causes which occur as a result of consequences. The requirement of introducing SPC tools of this nature is to remove non-conformity and improve quality in all stages which includes operational procedures and activities in a production process [27] [28].

The following figure shows a graphical representation of a control chart where the **horizontal axis** represents different time intervals when qualitative values were statistically sampled whilst the **vertical axis** contains obtained appropriate sample characteristic values. A criterion for comparison of sample characteristic values is done by calculating **control limits** [29].

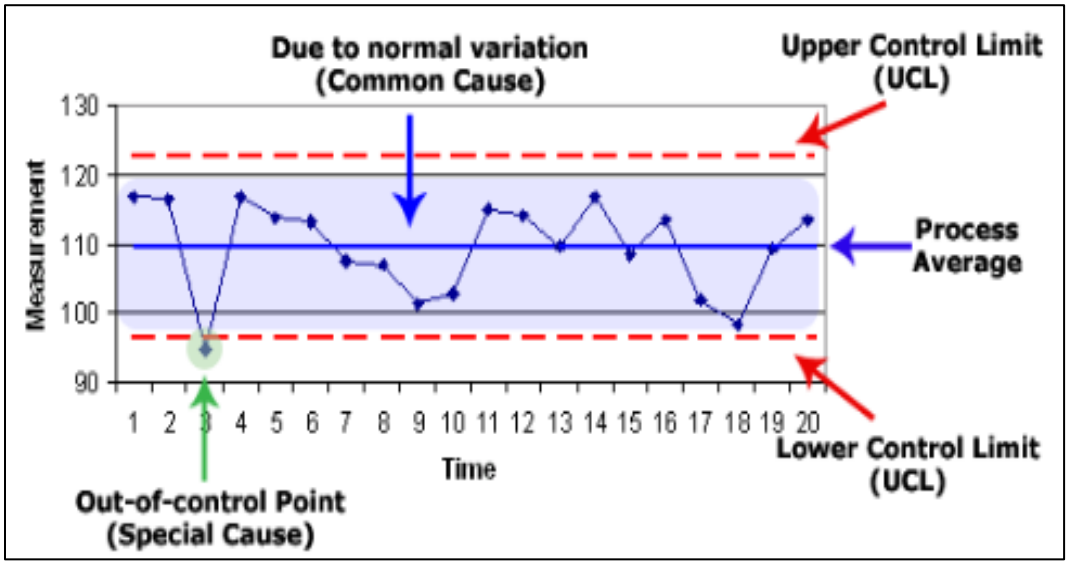


Figure 3: Control Chart representation with limits [27]

Upper Control Limit (UCL) is given by equation 1:

$$UCL = \bar{x} + 3 \times \bar{\sigma} \qquad \textit{Equation 2.1}$$

Lower Control Limit (LCL) is given by equation 2:

$$\mathrm{UCL} = \bar{\mathrm{x}} - 3 \times \bar{\sigma}$$

Central Limit (CL) / Average is given by equation 3:

$$\bar{\bar{x}} = \frac{1}{k}(\bar{x}_1 + \bar{x}_2 + \bar{x}_k) = \frac{1}{k}\sum_{i=1}^{n} x_i$$

2.2.4.2 Classification of Control Charts

1. **Shewhart control chart:** Is representation of a product characteristic extracted from *n* samples against time *t* or sample figure. A univariate type of Shewhart control chart as shown in *Fig 3* has the central, upper control and lower control limits with the latter two limits known as three-sigma limits. Main goal is to control the process mean whether it has deviated from our regulated value. By applying estimation using appropriate estimators, parameters of mean and variance can be detected if they have shifted or remained within control limits. Absence of assignable causes reflects that the process is in-control, and no further action is needed. Occurrence of non-random patterns on control charts in form of trends, cycles stratifications should signal an alarm when they fall outside control limits. Performance evaluation is done through Average Run Lengths (ARL) and quality engineers are able to establish run rules for quality control plans to either reject or accept process variations [28].
2. **Cumulative Sum control chart (CUSUM):** Is a sequential theory centered approach. CUSUM control charts unlike Shewhart, detect smaller shifts in process average. Calculations based on this chart provides a worker inclusive environment in the production floor. The modus operandi of this chart involves collecting sample from the production process at certain intervals in months/ hours/shifts. The main purpose of this chart is to reduce the Average Run Length (1) of an out-of-control at the same time maintaining the Average Run Length (2) or a process in-control.[30].

3. **Exponentially Weighted Moving Average control chart (EWMA):** This chart was established around the 1950s. The primary purpose is to detect minimal gradual shifts in process mean because of their ability to utilize data from a long pattern of samples. They are regarded as sensitive. CUSUM and Shewhart are traditional forms of control charts hence cannot detect these small shifts [31]. The Average Run Length (ARL) of statistically out-of-control process has an optimal value that easily gets affected by gradual shifts, which renders an EWMA chart greater ability to include shifts in standard deviation. Easy to adapt to automated data and is not affected by absence of natural assumptions [32].

2.2.4.3 Process Capability Analysis

Is a statistical control approach in Statistical Process Control, to prove whether a subprocess from a production line is capable of producing a product with required tolerance limits. Initial stage in Process Capability Analysis (PCA) data collection that is to find the current status quo process capability known based on selected standard deviation. Samples or components are drawn from the process, tolerance limits are then established. Process performance indices (**Ppk** and **Pp** values) are calculated, if they are found incapable of meeting desired specifications or customer requirements it calls for further investigation of the process. Data should be normally distributed or have passed a normality test; a histogram is employed before making capability calculations. For a capable process, it will be evident that product tolerances are within the specified limits. To ensure system stability, parameters below the capability index are subjected to Shewhart Control Charts to focus on elimination of non-random causes of variability [33, 34].

The formula below shows how to calculate performance indices:

Cp Index = $(USL - LSL) / 6\sigma$ (1)

Cpk Index = $\min(\mu - LSL, USL - \mu)/ 3\sigma$ (2)

Cp= 1: Process is minimally capable because process variability exactly meets specifications. Seemingly fewer defects on the process because 99.74 has fallen within limits, but in fact when calculated in Parts Per Million (ppm) defective products are high; that 2600 defects per million. Hence process capability needs to be increased. A different result comes out when we use Cpk performance index.

Cp≤ 1: Process variability is outside specification range. Calls for further improvement because the process is incapable of producing within tolerance limits.

Cp≥ 1: Minimal capability of process is exceeded. More tight process variability than specification limits.

where USL and LSL denote Upper Specification Limit and Lower Specification Limit respectively, σ denotes the standard deviation, and μ denotes the process average. Cp index instills that for a production process to produce within required requirements / a capable 6 sigma process, process capability Cp; should be above 2 and that USL-LSL must be less than half of UCL-LCL. [35].

2.2.5 Advantages of Implementing SPC in Industry

Implementing SPC is crucial because it enhances production competence by lowering scraps and rework and increasing performance by lowering variability. This researcher claims that in an effort to maintain competitiveness , US businesses had started using Total Quality Management (TQM) approaches such as SPC which leads to reduced unpredictability and flaws to produce a product of higher quality [36]. The majority of production and quality costs that SPC seeks to minimize are quantifiable. Resource size of a company has no bearing on success or failure of SPC application; but instead, appropriate planning and workers responsiveness to problem solving.

The benefits of using SPC can be categorized subsequently to uphold the necessary level of design conformance, raise product quality, do away with unnecessary inspections, reduce the proportionality of faulty parts acquired from suppliers, reduce customer returns, reduce scrap and rework rates, provides proof of quality thus making it easier to identify trends and the capacity to cut lead times and expenses [37].

SPC monitors manufacturing processes using statistical techniques in an effort to reduce variance, maintain and enhance product quality. SPC are usually limited to one production stage in the process in industry.

An application of SPC in fishing preparation industries employed simple systems. Such a system deducted the number of giveaways and unwarranted rejections during check-weighing after controlling of package weights. This is evident of good manufacturing practices as all documentation, or records needed for trading standard inspections can be kept. Such an

organization can easily aid authorities with routine inspections by providing high quality records in form of data sheets and control charts [38].

The use of SPC technique was also found in tile industries. They recommended using SPC to cut down on waste and undesired ceramic tile flaws. They used a pareto chart to identify the causes for unsuitable consequences and they classified them. R, X and range control charts[39] were implemented for handling variations in the process. Additionally, they embraced dispersion diagram to assess and contrast the performances of over time. The process is deemed capable. Implementation of SPC tools correctly in the production plant minimized ceramic tile wastage[40].

Another effective application of SPC techniques was in a company which manufactures glass bottles, was in a processing line to lower defect numbers by locating where highest waste is occurring. A direct examination of the production line was done. Information was collected through questionnaires from company employees and potential customers. Pareto chart which works on a principal law of 80/20%, to identify the 20% vital few problems and p-control chart was implemented. The main goal was to educate and equip the quality team on how to exploit process data, conduct effective brainstorming sessions and represent this data on pareto charts. They discovered that the melting processing line was responsible for highest waste due to trickle. The forming line caused losses due to defective products rejection. Vital few issues were identified as pressure failure, overweight, blisters etc. Root cause analysis identified problems such as foreign matter contamination, improper machine maintenance, lack of process control skills etc. Despite having to implement more suggestions towards improvement, the company embraced significant productivity that will come with it [41].

Advantages related to the effective use of control charts (construction, information, interpretation) in industries was explored. Primary groups in terms of man, machines, materials, methods and environment/milieu, referred to as 5M method, were introduced. Application of SPC was divided into three main stages namely understanding the process, which is derived by process mapping, use of control charts in identifying sources of variation and finally how to eliminate sources of special assignable causes of variation. By implementing these stages in SPC, product quality can be enhanced in industries and product costs lowered [42].

A number of software tools can be applied together with SPC techniques. The importance of incorporating SPC in software development procedure was highlighted. Frequently used software's in industry now are ZONTEC (stability and capability of manufacturing systems), MINITAB (for real time data collection, performance metrics, corrective actions), WinSPC (OPC data collection, point-of -production analysis), XLSTAT (for data entry, data linking, data management) and SCADA. Organizations are encouraged to get the software package matching their specifications and requirements[43, 44].

2.2.6 Limitations in SPC application in Industries

1. Process control techniques in multivariate processes/data can pick the presence of outliers but face difficulties in identifying variables responsible for the problem occurrence. Rigorous mathematical calculations and statistical approaches exists. Data mining, machine learning and multivariate Statistical Process Control are research topics trying to solve the problem [45].
2. Traditional SPC charts are based on that process observations are identically and independent distributed (normal). These assumptions cannot satisfy recent applications involving complicated data structure (big data) and existing SPC methods [46].
3. Lack of proper training or expertise to employees on implementation of SPC tools.
4. Unavailability of reliable data collection system and measurement systems.
5. Lack of management support and employee empowerment and resistance to SPC as process improvement technique.

2.3 Anomaly Detection in Manufacturing

Anomaly detection is a field where computer vision is responsible for identifying rare occurrences in form of events, points in data and observations from a normal data set behavior. Industrial engineers are trying their best in improving methods in detecting anomalies in time-series data. Its application has emerged dominant in utilities industries, health care systems, security, risk and is involved in roles of data mining, computer-based vision, machine learning and statistical processes. Anomalies are usually in form of noise, faults, damage, outliers, novelty, peculiarities etc. Implementation of a system capable of identifying irregularities enables administrators in reducing losses [47, 48].

Normality is related to following **Gaussian behavior** in form of common causes of variation and anomality/ abnormality is deviation from randomness or specific causes of variation.

2.3.1 Types of Anomalies

In the context of smart manufacturing systems, an anomaly is defined as an unexpected change in behavior or a status of an Internet of Things (IoT) behaving outside the expected norm. Normal procedural behavior of any system can change at any time for different reasons. Time series anomalies are divided into point, contextual and collective anomalies [49].

2.4 Machine Learning in Anomaly Detection

Strategies in anomaly detection are usually divided into stages of training data function to build the model; this includes supervised, unsupervised and semi-supervised anomaly methods. Machine Learning is one the techniques gaining visibility in detecting anomalies. Machine Learning make use of algorithms in analyzing large volume data sets. Therefore, the essence of supervised anomaly detection problem [50].

2.4.1 Supervised Anomaly Detection

When building predictive models using supervised form of AD, labels are required for each data observation. Availability of information to supervised methods makes them have a better detection rate. On a more accurate note, they are marred with technical challenges that discredits their accuracy which are lack of training data set coverage and high false alarm rate caused by failure in obtaining in acquiring authentic labels. The reason for false alarms is the presence of noises in training data sets. Commonly used supervised methods are Bayesian Networks (Naïve Bayes classifier), Decision Tree, Supervised Neural Networks (SNN) and K-Nearest Neighbors(k-NN classifier) [51].

2.4.1.1 Supervised Neural Network (S-NN)

SNN learning is concerned on behavioral and user prediction. Have a greater capacity in dealing with many challenges based on rule-based techniques. A greater advantage of Neural Networks is the capability of handling imprecise data and uncertain information. They do not rely on patterns of historical data in order to conclude solutions in data set.

An example of SNN is Artificial Neural Networks (ANN) is a network with massive interconnections and parallel processing structures. It has gained usability because of its high capability in establishing nonlinear correlations between variables even when input is unknown. ANN models are data driven and resolves challenges through machine learning neurons [52].

A stochastic gradient descent optimization algorithm (SGDA) is used in training an NN model. Through the backpropagation algorithm weights are updated, deriving from loss function errors. Responsibility of the loss function is to calculate differences in actual output/ground truth and predicted output and their application depends on which type of problem being dealt with. So, the main of goal of the backpropagation algorithm (supervised learning) optimizes the loss function by updating weight of NN. Forward propagation algorithm feeds input data to for output generation [53].

K -Nearest Neighbor (k-NN): This is a neural network in non-parametric form. Its main task is to determine rough distances between various point on input vectors as classification. In the process, unlabeled points are moved towards the class of nearest K-neighbors. The main advantage of this method is easiness in implementation and adaptable in both regression and classification problems. Poor performance of KNN is experienced when dealing with unbalanced samples of data hence an increased complexity for large datasets [54].

Support Vector Machines (SVM): SVM make use on classification on training vectors. This is achieved by mapping of input vectors into a higher feature space to obtain an optimal hyperplane. Training samples are not responsible for determining the hyperplane making it more adaptable to outlier classification. They are advantaged in dealing with high dimensional data. To be able to execute all these functions, they require a good kernel functions, a lot of memory and CPU time [55].

Recurrent Neural Network (RNN)

RNN makes use of modules as memory storage for delicate information from previous steps. Their distinct different from feedforward neural networks is that they exhibit feedback loops hence can accept input sequences. RNN operates in sequential processing pattern with recurrent units into a full network. It works through backpropagating at time (BPTT) when feedback process is modified weights are updated. BPTT is prone to network instability due to increased error gradients when updating NN model weights [56, 57].

Long Short-Term Memory (LSTM) Neural Network

This is a model that overrides problems associated with R-NN structure. They handle time-series data that have long range persistence. Figure 4 below shows the structure of an LSTM.

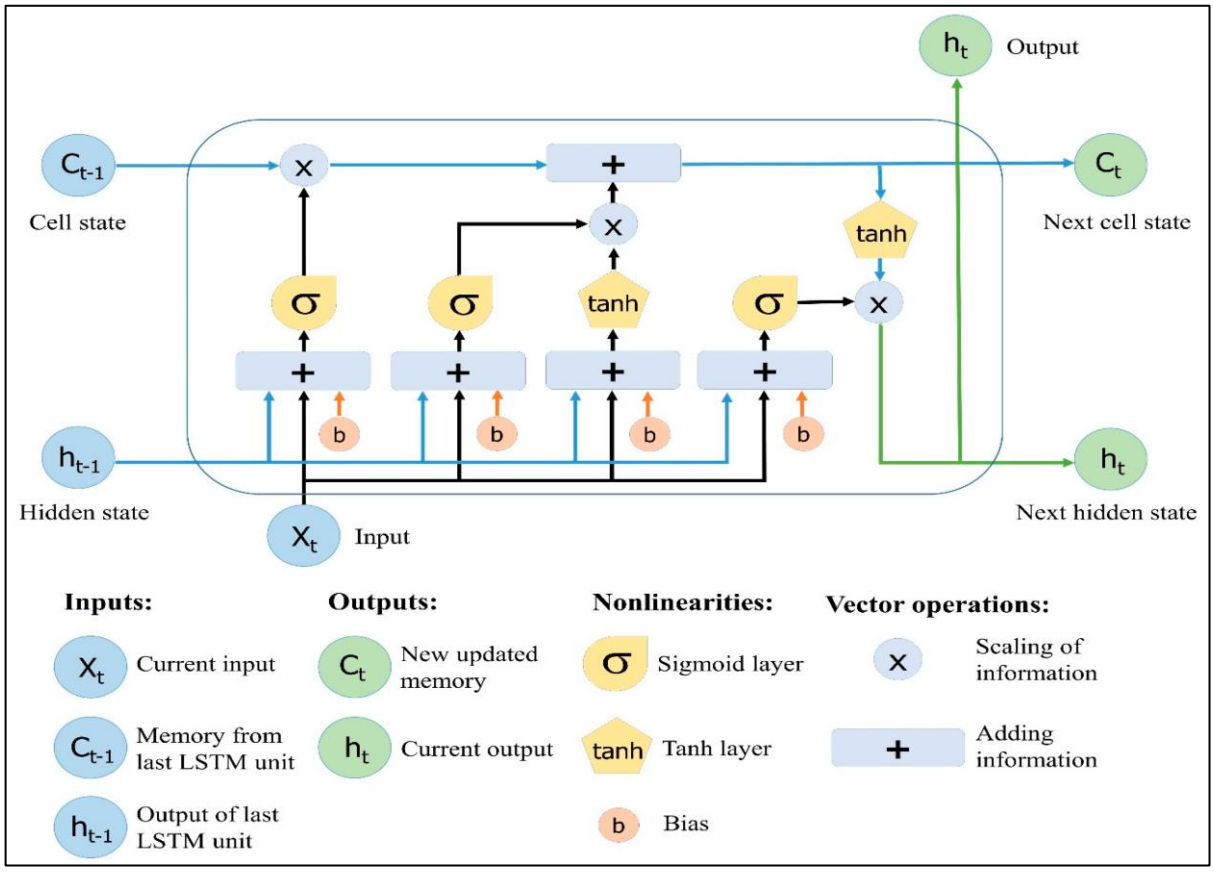


Figure 4: Diagram representing Long Short-Term Memory (LSTM) Neural Network [58]

Data flows forward unchanged in a in a cell state. Sigmond gates can add or remove data from a cell state via linear transformations. The main focus is to get rid of long-term dependencies through the use gates (gates contain matrix operations and different weights). Information that is deemed unnecessary, is identified and then excluded from the cell through the sigmoid function [59].

LSTM networks have been known for their greater accuracy in prediction tasks than most of the neural networks. The greater advantages associated with supervised AD methods is their algorithm simplicity, easiness to understand. Large memory requirements means time is going to be consumed hence labeling of data is almost impossible [60].

2.4.2 Semi-supervised Anomaly Detection

This class of anomaly detection has the goal of distinguishing occurrences that are behaving in a minority manner than the rest. There is availability of normal occurrences in the semi-supervised training sets. Labels are not a necessity requirement. Real life applications of semi-supervised techniques include detecting of fraud activities, identification of fake news on public platforms and in medical field [61].

2.4.3 Unsupervised Anomaly Detection

The application of machine learning in unsupervised AD techniques the assumption lies in that we are working with normal data samples and a small portion of the samples follows abnormal behavior. So, data from a majority portion is considered normal whilst the other one is considered an attack and unwanted. It is not necessary to have a set of training data in this method. Examples include self-organized maps and single class support vector machines [62].

A typical application of **Density-based outlier** detection techniques (unsupervised) starts with assumption that outliers denote expected anomalies. We then use this technique to find the density of data spreading around data points to find anomalies. Another approach implements the **local outlier factor,** an outlier score**,** which averages the k-nearest data points and local density of the original point [63].

Unsupervised learning techniques carries a greater advantage in that no label data is needed and their application is wide. Whereas a bit of discredit has to do with relying on assumptions that normal incidents are more frequent than abnormal ones.

2.5 Challenges of Anomaly Detection

The identification of behavior that is not following Gaussian behavior has proved to be a difficult task, since it is crucial to identify what is being defined as a normal and what is being termed abnormal behavior. The literature reviewed was able to pick up the problems below [64] [65]:

1. **Normal Region:** It is difficult to properly arrange a method to describe normal and abnormal occurrences.
2. **Identifying attacks:** It has been discovered that many unwarranted attacks can pose themselves as normal instances.
3. **Behavioral changes:** Sometimes it is possible for the current normal instance to change to other pattern, then there is need to redefine normal behavior criterion.

4. **Data challenges:** The dependence on labeled data for training pose a great challenge to most AD techniques since it cannot always be available.

2.6 Link between SPC and ADM

In this section, a possible link between SPC and ADM is explored. This of major importance in accessing process behavior ability to conform to required specifications and also identifying major trends within process variables. The summary below is based on the authors knowledge to SPC tools in quality management and currently reviewed literature on AD [66]:

SPC is a technique used in AD in industries for the main purpose of monitoring or controlling and reduction of variation in a process, in the minimum time possible in pursuit of a high product quality at minimal costs. SPC approach happens within specified limit interval. In the context of process control, Anomaly Detection identifies instances (in form of anomalies or outliers) that do not conform to normal process behavior.

SPC employs a fundamental tool called a Shewhart Control Chart to monitor various process parameters or characteristics (process stability). This is a key process in controlling manufacturing or production of products by implementing corrective actions timely in order to keep our process centered and maintaining variation within specified limits. In this instance, AD points out shifting behavior of the process and can be rectified using SPC techniques that is quality control tools.

Use of traditional control charts in SPC faces difficulties in interpretation, trend observation and even in designing them. The underlying assumption(s) is that process samples are normally and identically distributed and historical data estimates values/size of main parameters. However, this kind of approach overlook issues to do with autocorrelation of data in complex natures (yet a variety of variables exist already with covariant correlations) and cannot fit some dynamic behavioral conditions in industrial processes. This has led to emergence of false alarms in production processes. Deployment of Machine Learning and Anomaly Detection counteracts most problems hence a data driven decision making approach to produced advanced control charts.

SPC main task is to detect common-cause variations (freak patterns, trend, shifts, cycles etc.) and assignable-cause variation (human errors, calibration issues, process parameter change etc.) in order to declare a process in-control or out-of-control. In SPC, out-of-signals are identified

and then eliminated. This approach does not allow us to observe and monitor abnormal behaviors to indicate future references or to begin potential root cause analysis. This becomes a challenge in SPC pattern recognition. However, the introduction of AD, Machine Learning has collaborated well with automation in amassing big data parameters for possible monitoring for example (sensory information, process variables, historical data) through sensors in Smart Manufacturing in identifying unreal trends and extract features from control charts.

2.7 Research Gaps and Trends

Available literature on well recognized global sites have left a lot of scientific gaps that can be explored for future directions. Papers below summarized important aspects in Statistical Process Control, Anomaly Detection in Manufacturing and Machine Learning in Predictive Quality Management.

Sr	Papers explored	Research gaps
	Statistical Process Control (SPC)	
1.	A systematic comparison of PCA-based Statistical Process Monitoring methods for high dimensional, time dependent processes [67].	Lack of research towards adaptive and non-adaptive methodologies in handling process deviations on real time basis.
2.	Multivariate Statistical Process Control with Industrial Applications [68].	The need to explore more multivariate SPC techniques for high dimensional data. For Hotelling T^2 approach we face difficulties as dimension increases, assumptions on stationarity and normality.
3.	Quality Control for Smart Manufacturing in Industry 5.0 [69].	The need to explore human-machine co-creativity frameworks and standards in order to integrate with evolving new technologies.
	Anomaly Detection (AD)	
4.	A Methodology for evaluating the robustness of Anomaly Detection to Adversarial Attacks in Industries [70].	Vulnerability to adversarial attacks is degrading prediction performance for most systems. Need to venture into more robust defense mechanisms.
5.	Interpretability-aware Industrial Anomaly Detection using Autoencoders [71].	Direct explanation of results for predictive models enhances reliability and enhances decision making.
	Machine Learning for Predictive Quality Management (ML in PQM)	
8.	Continuous improvement and adaptation of predictive models in smart manufacturing and model management [72].	The need to have proper documentation of prediction models data, in trying to adapt to everchanging software environments.

Table 2:Identification of research gaps and scientific trends.

2.8 Summary of Literature Review

The above literature review provided a comprehensive investigation of publications, scholarly material in our main research of Machine Learning in Predictive Quality Management. Research gaps and scientific trends were also identified for future direction research in main fields of Statistical Process Control, Anomaly Detection in Manufacturing and Machine Learning algorithms. The highlighted areas gave a leeway on which direction to take as we evaluate our problem statement into feasible solutions.

CHAPTER 3: Methodology

3 Introduction

The author describes the methods used to carry out this design project in this section. Addressing the research objectives described in earlier introduction, this chapter intends to present the study methodology by focusing on detailed description of research design, data collection process and different statistical or qualitative techniques methods for data analysis. The chapter goes on to discuss the planned data presentation and analysis, as well as the quality and dependability of data obtained.

3.1 Research Philosophy

A phenomenon known as research philosophy explains the origins, nature and progression of knowledge. Another way to look at it is as more of an expansion of the methods used to gather data when conducting research. Most scientific or quantitative research use positivism as a conceptual foundation. Since positivists value and prefer empirical hypothesis testing, quantitative research typically lags behind this approach. Simply gathering and analyzing numerical data is what quantitative research is all about. The author decides to employ positivist philosophy in this study since it takes a quantitative approach. A positivist research paradigm served as a researchers guide in this study and systematic research was used to carry out the requirements of research objectives [73]. For acceptance in fisheries industry, the prediction of fish location or fishing effort and breeding patterns using machine learning methods needs to be assessed. The evaluation makes statistical analysis a quantifiable step in achieving the study's key goals.

3.2 Research Design

By logically and methodically combining the many components of a study, which makes the research more effective in solving the research problem. A study design primarily includes the blueprint for data collecting, measurement and analysis. The design of case study research typically incorporates qualitative and quantitative approaches and is used to undertake in-depth research on subject of interest in its natural real-life context. Use of descriptive research and explanatory research (prediction of future occurrences) was implemented [74]. Furthermore, to facilitate easy analysis and identification of patterns in the results and the drawing of factual conclusions, the data collected was expressed with quantitative data presentations. Phases 1 and 2 of this report are complete. The creation of a machine learning multi-objective optimization algorithm is of phase 2.

3.2.1 Methodology Layout

The below pictorial view clearly shows the different phases as we develop our project idea into a fully-fledged product ready for launch.

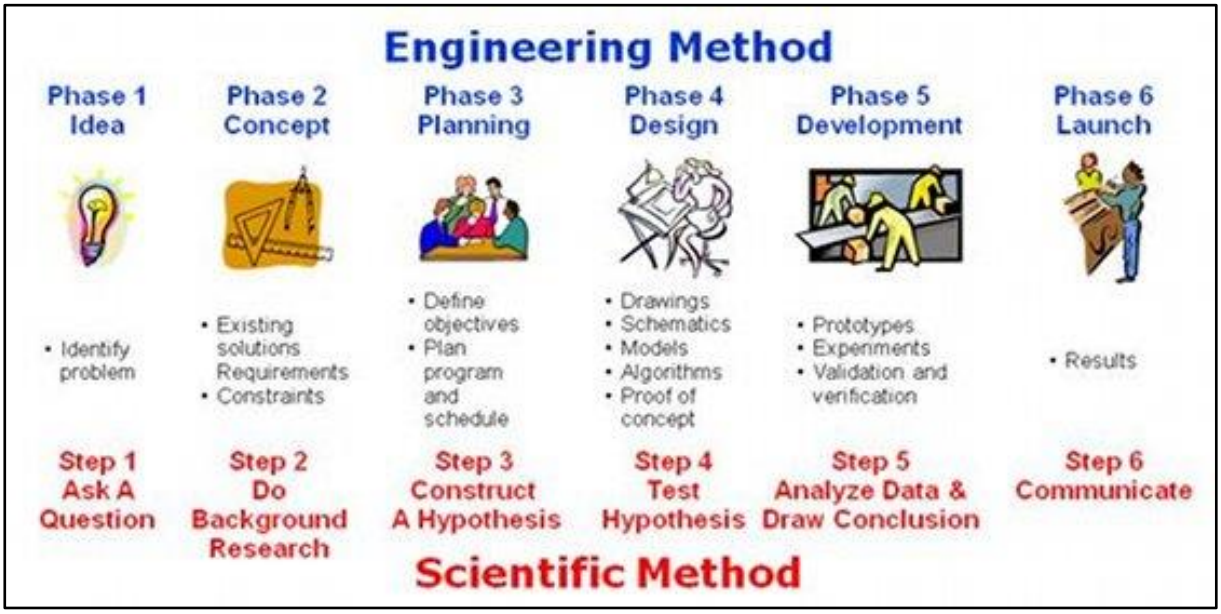


Figure 5 : Overview of Scientific Method approach

3.2.1.1 Idea

Typically, an issue comes first in the idea phase. Usually, the problem statement is only loosely defined, necessitating investigation (asking questions like how, why?) into its viability and practicality. In order to pursue the answer, there must be a significant value (or demand in case of product development). The author of this study determined that the utilization of data gathered to predict fish movement, breeding patterns and likely yields always needs to be improved.

3.2.1.2 Concept

During the concept phase, a diverse range of models, including mathematical, physical, and sketches are generated. This entails reviewing existing literature about the key subject areas in our case machine learning algorithms, statistical process control, and anomaly detection. The authors evaluation of alternative options, informed by discoveries of other scholars, constraints faced by existing solutions which further establishes the research gap of this study.

3.2.1.3 Planning

During the planning phase, the implementation strategy is formulated, clear project objectives are set, plan of actions and schedules established.

3.2.1.4 Design

The crux of the matter lies in the design stage. Specifications are established, and details are given. Both this phase and the development phase are referred to as “design planning” and “detailed design” respectively. The objective of this phase is to translate the systems engineering model and client requirements into engineering specifications in form drawings, schematics, exploring the selected algorithms that an engineer may use to create and construct functional prototypes.

3.2.1.5 Development

A physical prototype or a virtual working simulation may provide a viable solution. Given in our problem question, the regression algorithm models are trained and tested with data. Performance evaluation is done through evaluation metrics considered to be ideal with the nature of the task.

3.2.1.6 Launch

In this report, the results drawn out form the performance of our models are recommended to authorities following data analysis about the deployment of multi-objective optimized prediction model based on machine learning are considered as the launch aspect.

3.3 Research Population

Employees at fishing companies in Norway served as co-researchers study population in our case. As a result, the research obtained most of its data from websites responsible for keeping catch data, AIS data, environmental data and landing data. Participants with experience and expertise of vessel fishing were also consulted.

3.3.1 Purposive sampling

By applying the typical case and expert sampling techniques of “purposeful sampling” the author interacted with experts from Kystverket and Global fishing watch and recommendations from people who have embarked on similar projects.

3.3.2 Sample size

Sampling is a crucial statistical procedure that involves the systematic selection of a predetermined number of observations from a larger population for analysis.

3.4 Validity and dependability of product

It is important that a defined criterion for quality standards is put in place which is unit of measure for the integrity and validity for the resultant product. We can reliably depend on a product, if error tolerances are very low and results produced are of high standards [75]. This is the purpose of doing a quality evaluation our algorithms performance.

3.5 Presentation of the data

The results from data visualization and descriptive statistics from our dataset behavior were analyzed and the researcher presented the data using graphs, tables and bar graphs. A narrative description of the data and created tables, graphs and charts was written to reach conclusions from the statistical analysis of our data.

3.6 Ethical considerations for the survey

Participants required reassurance that the information gathered about them would be kept confidential and kept anonymous. The volunteers could not have been coerced to provide answers and relevant data by the researcher. The researcher had to follow aforementioned ethics for the study to be successful. The ethics of data collection and interactions with respondents were observed.

3.7 Case Study

Chapter 1 of this Thesis summarizes the problem statement as given by a fishing company Oddvar Nes; in Senja, Norway. They run two coastal spinners as their main fishing vessels. Over a long period, they have been gathering “big marine data” from relevant data sources in form of AIS data, environmental data and historical catch data to be brief. Complexity of big marine data needs fishing companies to establish correlation relationships in order to identify trends and patterns towards sustainable fishing practices. This in turn reduces operational costs in form of fuel expenses, salaries etc. This thesis seeks to answer how machine learning algorithms have a greater advantage over traditional Statistical Process Control (SPC) methods predicting the likely fish location or availability in sea. Machine Learning algorithms are well suited for handling and analyzing large datasets including environmental conditions, information on fish species and relevant data in fisheries. They can adapt well to varying marine data including fish movements, changes in seasons and influencing factors like human activities in the sea. This project further seeks to address these disparities in order to provide the company with recommendations suitable for decision making towards a sustainable fisheries management.

3.8 Engineering tool for concept selection

There are numerous MCDM techniques available to help in concept selection. Each of these approaches has its own unique viewpoint for determining the worth, value and merit of the compared notion alternatives. In this work, Pugh's concept selection was used due to its intricacy and iterative methodology. The technique puts the criteria on the vertical axis by using one leg the matrix.

3.8.1 Conceptual Framework

Machine learning has greatly developed its application in large-water datasets aiding analysis in (a) *evaluation and findings* inclusive of object detection and categorizing of fish species in sea, tracking and navigation of species, seabed mapping analysis (b) *management and monitoring of marine data* including application oil and gas industry in detecting corrosion of pipelines, setting up warning systems in case of potential disasters (floods, tsunamis) and monitoring of environmental data or parameters like ammonia concentrations, quality and pollution analysis.

Prediction of most oceanic instances have been modeled and formulated as regression problems hence the application of deep learning techniques, machine learning models and statistical based models in analyzing marine data.

Figure 6 below is an architecture that aids as the basis in formulating predictive models' concepts in pursuit of drawing decisions and knowledge from the output results of the process.

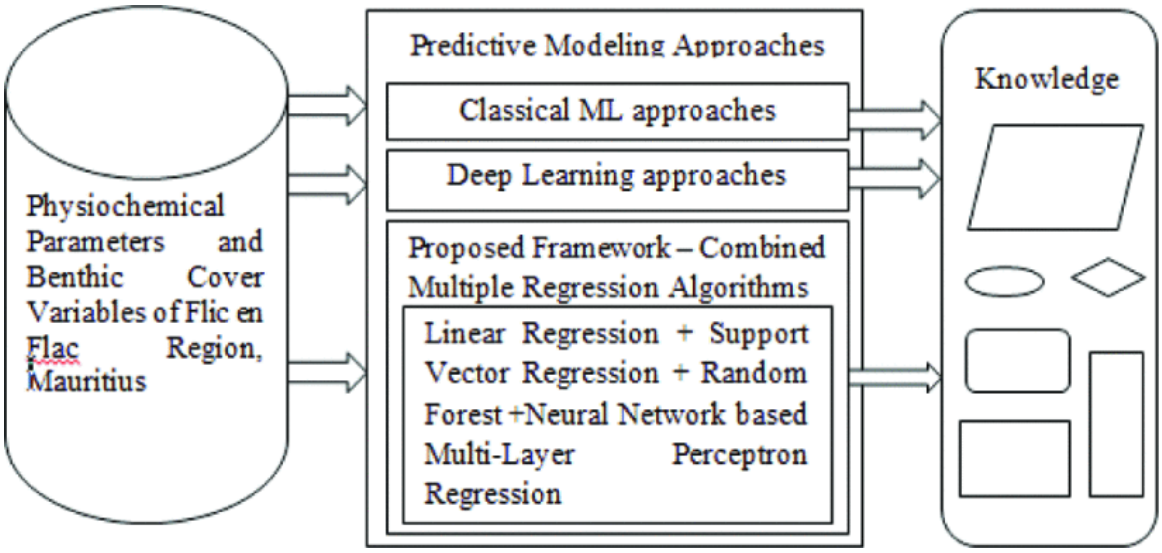


Figure 6: Overall conceptual framework(([76]

3.8.2 Concept 1: Multiple regression algorithms for fish movement patterns (location) or fish abundance in sea

For prediction of fish abundances or location to have a higher yield, the incorporation of machine learning algorithms has to be goal driven whilst emanating from existing problems. The concept of AI in this study is derived from predictive data analytics. Multiple regression algorithms inclusive of (a) Decision Trees (DT) (b) Random Forest (RF) (c) Linear Regression (LR) (d) Support Vector Machines (SVM) and (e) Gradient Boosting Regression (GBR) are employed in this concept.

Development of this concept was done following the criterion mentioned below:

1. ***Establishment of the goals***

The goal(s) to be reached by incorporating regression or classification algorithms is to have accurate and precise prediction of location of fish at any given time.

2. ***Data collection***

Information in form catch data was gathered from previous records from a fishing company. External data sources/ relevant sites were consulted to produce AIS data and environmental data and corresponding yields in previous years. Data can be in structured or ordered form, usually presented in excel spreadsheets, comma-separated values (CSV) and structured query language (SQL) formats.

3. Data cleaning and preparation

For AI algorithms to be able to analyze data efficiently, it needs to be free of mistakes, redundancy data and omitted values. Data cleaning involves removing duplicates, adding omitted values and fixing errors. The objective of this preprocessing stage is to pave way for clearer and accurate data hence more precise analysis. Since we are dealing with a big marine data (AIS data + environmental data + catch data), datasets can be structured, unstructured and semi-structured and for them to be in consistent format with machine learning algorithms data preparation needs to be done.

4. AI algorithm selection

Since this study focuses on multi-variable inputs to give a prediction, machine learning algorithms like classification and regression were selected based on different criterion like dimensionality of dataset, problem type, scalability, and robustness of the models. ML algorithms selected for this concept have been highlighted above.

5. Data Modelling

The data collected in (2) and prepared in (3) is suited to suit the machine learning techniques selected in (4). In this study, SVM, RF, K-NN, GBR and DT are used to generate forecasts or species abundance. The figure 7 below shows the summary of data modelling for this study.

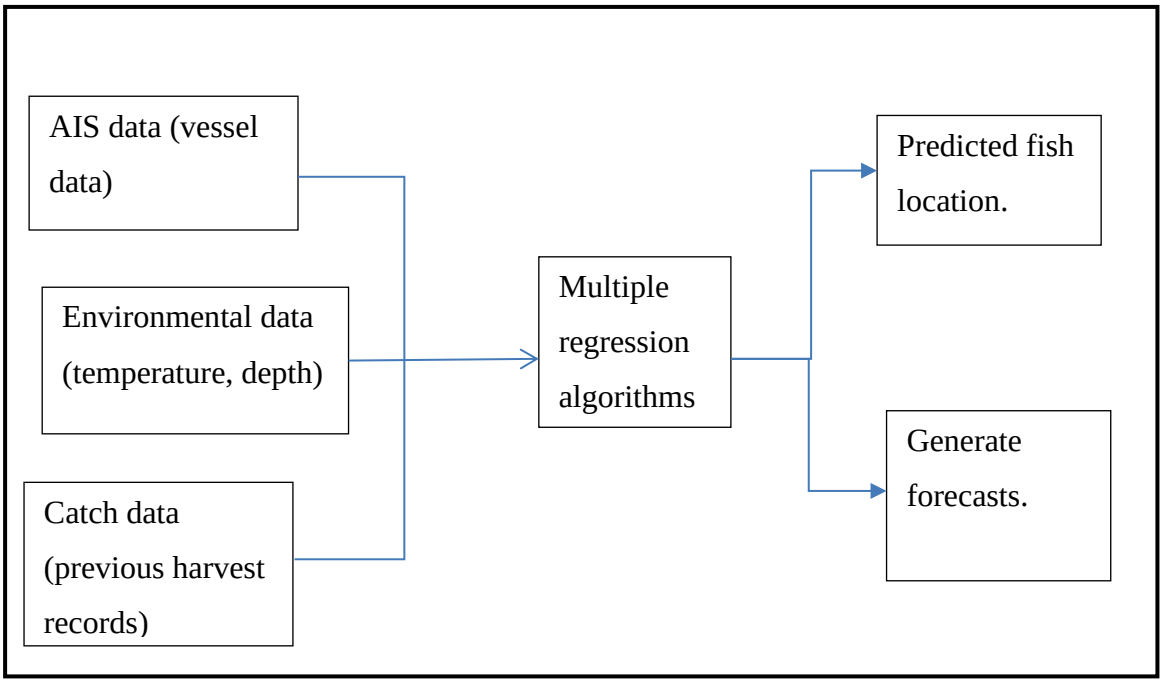


Figure 7: Multiple ML regression algorithms model

6. ***Integration with existing systems***

The predicted data which entails optimized data has to be used together with other organizational data management software. To achieve such, AI algorithm is linked with already existing software such that the results of AI data modeling can feed directly into existing process and hence achieve optimization.

7. ***Testing and validating the AI algorithm***

The result of the algorithms is tested by comparing predictions of several iterations with the reality of fishing using vessels in the sea. The performance metrics of machine learning models used in this concept to evaluate are as ROC analysis, Root Mean Square Error (RSME), Mean Absolute Error (MAE), accuracy and precision. These metrics assist us in making critical decisions on which algorithm(s) is dependable and performs better in predicting fish availability in sea and establishes relationship between static and predictor variables.

Advantages of concept 1

1. Each machine learning algorithm is given its chance to show case its capabilities in terms of handling large data sets.
2. We can identify existence of relationships (linear/non-linear) between predictor variables in question.

Disadvantages of concept 1

1. Some machine learning algorithms are affected with overfitting issues by outliers.
2. Variety of algorithms perform predictions differently due to minimal variations in training dataset. Computations become more intensive when large spatiotemporal dataset is used.

3.8.3 Concept 2: Ensemble learning technique: Prediction of fish species by capitalizing on strength of multiple machine learning models

In this concept, Multilayer Perceptron (MLP), Logistic Regression (LR) and K-Nearest Neighbors (K-NN) machine learning models are used in an ensemble learning approach to leverage on their individual strengths to produce a more accurate prediction performance [77]. The prediction in this concept is taken as a classification problem. This involves making some ML algorithms trained first on base learners. Further, their combined prediction output is trained on a metal learner machine algorithm in this case, logistic regression, to produce the final prediction. Formulation of this model has been greatly influenced by this research which was conducted in Manila Bay, Philippines [77].

3.8.3.1 Important Terms

1. ***Majority Voting* –** Both regression and classification tasks can aggregate their output from base learners towards increasing overall performance.
2. ***Bagging* –** Involves taking a single training set or bootstrap samples to train different sets of the same classifier and calculating its average.
3. ***Boosting* –** Combining of iteratively weakly learnt models in sequence.
4. ***Stacking* –** Training of meta learners from an aggregated output of base learners.

The dataset for this concept extracted attributes or variables that can be classified as predictor variables and independent variables. The fishing variables to be trained with the ensemble learning approach are:

Year – recorded year(s) for which fishing harvesting activities occurred.

Species- type of fish species of major interest

Landing area – refers to area where vessels are docking after a harvest to offload product.

Fishing gear - with relative to the fishing method used to catch the target species.

Fishing area – relative location or area where the species are caught.

Quantity – volume of production

Season – refers to the period where catches are high, low and unavailable for a species.

3.8.3.2 Stages in development of concept

1. **Pre- processing stage**

The K-NN means algorithm is used to cluster the available fisheries data in order to classify species in terms of availability (low, medium and high). Number of clusters (k) are determined first followed by normalization of dataset columns to improve performance, stability and interpretability of the clustered output. Further, you determine Euclidean distances between the k clustered points. Finally, the K-NN model is re-run for the purpose of meeting convergence and keep on updating center points.

2. **Ensemble Stage**

K-Nearest Neighbors, Multilayer Perceptron with backpropagation for better performance and Logistic Regression are used as base learners and their aggregated output through majority voting ensemble method produces new data predictions that classifies the specifies as either low, medium and high. Number of neurons, learning rate and the activation function are chosen for MLP which best suits the dataset in use. Secondly, the output through stacking ensemble learning model with a meta-learner, Logistic Regression algorithm, predicts the availability of species.

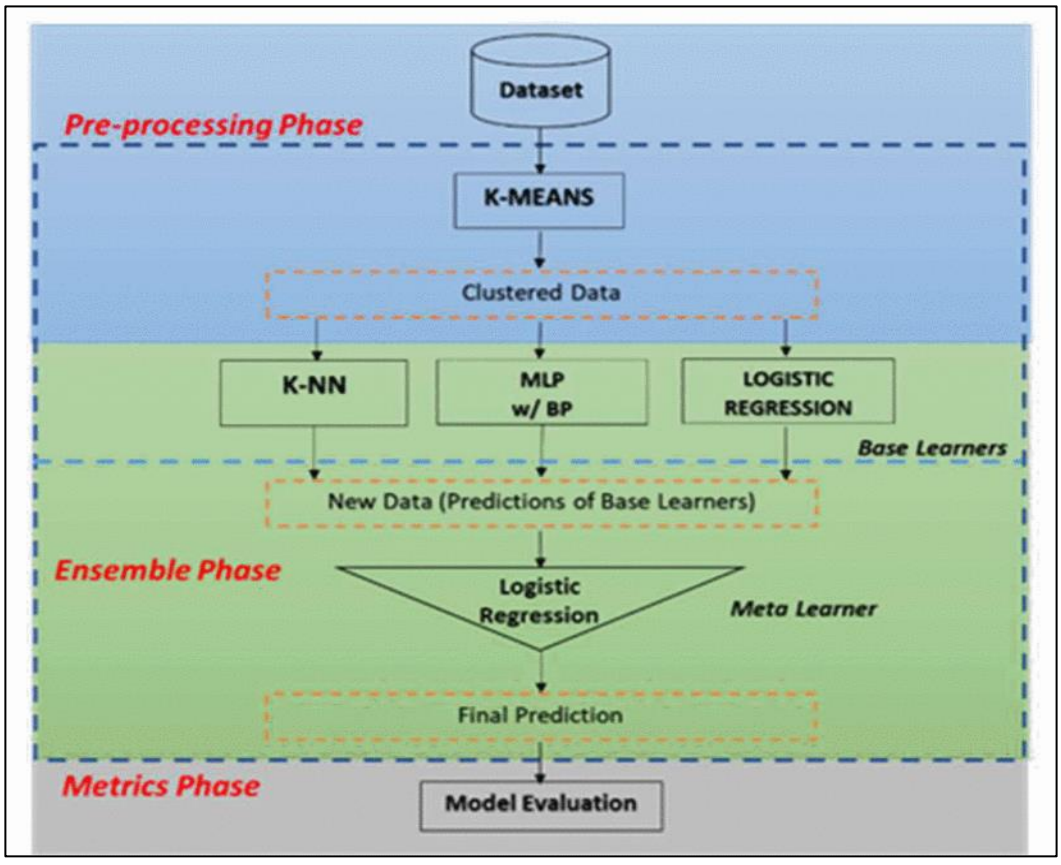


Figure 8: Adopted ensemble learning approach architecture [77]

3. **Performance evaluation stage**

Performance assessment of the ensemble learning model concept is done through various evaluation metrics which are AUC, F1 score, classification accuracy and recall. The predictive classes of the model are later incorporated into a confusion matrix to demonstrate the correct numbers of accurate and inaccurate predictions for each class up to the combined predictions (by majority voting) of each machine learning model in this case; K-NN, MLP and Logistic Regression.

Receiver Operating Characteristic (ROC) analysis: The target probability of or accuracy of the model is determined for correctly identifying the fish species which is the optimal classification threshold of the model.

Advantages of concept 2

1. Ensemble learning approaches capitalizes on the use of multiple learners' individual strengths to produce more accurate decisions.
2. Use of stacking maximizes model accuracy and enhances overall performance.

Disadvantages of concept 2

1. Ensemble learning approaches takes more time in training individual models with multiple base learners.
2. Implementation can be very complex from combination of multiple models, and this can lead to overfitting issues and makes the overall model less interpretable.

3.8.4 Concept 3: Prediction of spatiotemporal abundance of fish by forecasting sea bottom temperature

This concept makes use of deep learning model for predicting sea bottom temperature. The training is done from satellite images. A machine learning model, gradient boosting model, is incorporated with the environmental, landing data and VMS dataset from electronic logbooks. The area of interest is demarcated, and longitude& latitude range data is taken. On this particular region, data from previous years is collected of the bottom temperature in sea. Some of the necessary fishery's data are landing data, fishing gear, information on vessels.

The architecture and adoption of this concept fits well our research question and adjustments were done accordingly and can be summarized in figure 9 below [78]**:**

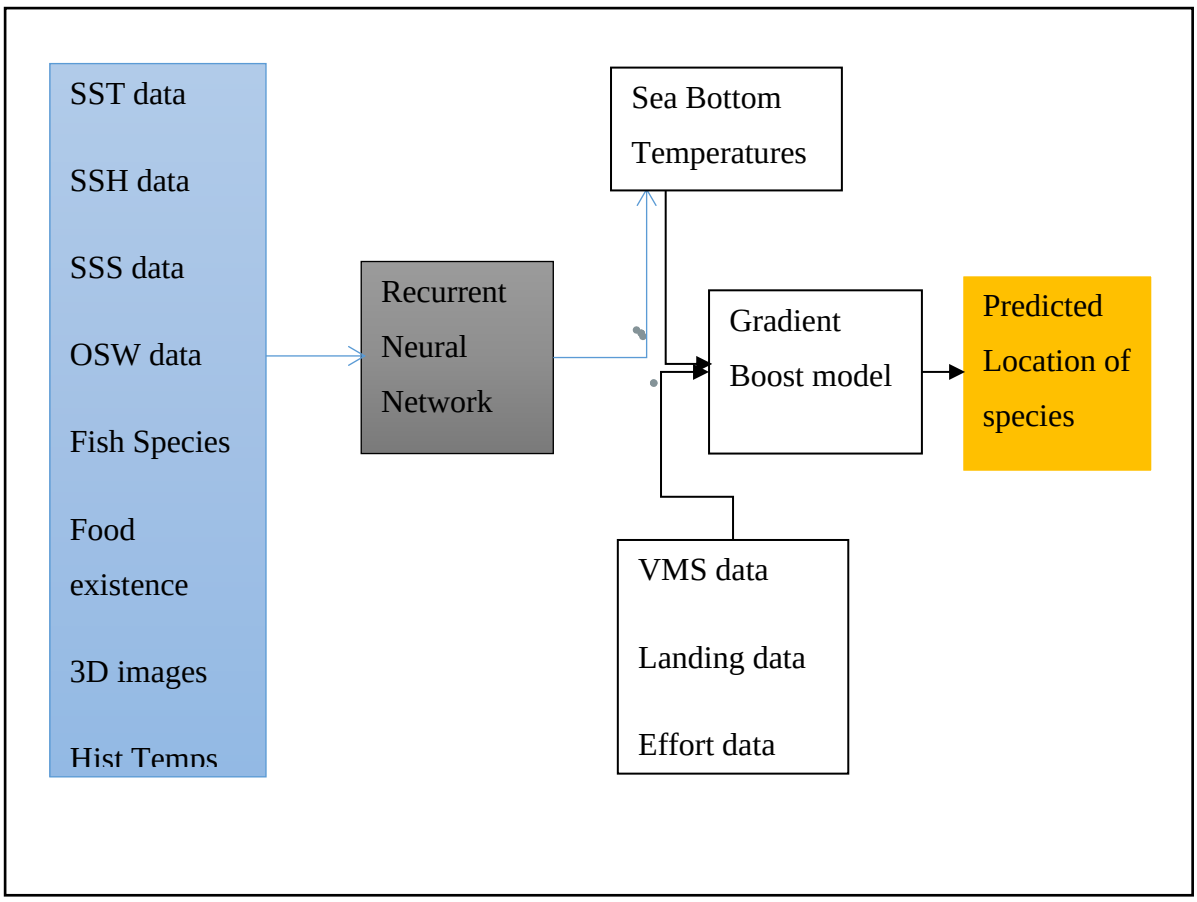


Figure 9: Architecture for combined forecasting and prediction model

Satellite data incorporated with features that aid in detecting sea bottom temperature are:

Surface Temperature (ST)

Surface Height (SH)

Surface Salinity (SS)

Sea Surface Wind (SSW).

3.8.4.1 Stages in development of concept

1. **Sea Bottom Temperature Prediction**

On this stage, a series of parameters including (ST, SH, SS, SSW) that contribute to temperature variations, for certain longitude and latitude space, amongst others are fed in a deep learning technique, Recurrent-Neural Network in form of Convolution LSTM to produce a sequence of consecutive bottom temperatures.

2. **Gathering of fisheries data**

Datasets of fishing vessels (VMS data) in form tonnage, vessel trajectories, distances and time spent together with landing data in form historical land data, catch rates and finally effort data in form of gear type, day, hours and mesh size are merged to form a high spatiotemporal dataset with *N* observations ranging from 2010- 2023.

3. **Gradient boost modelling**

A Light GBM is used to build a model for the specie in question, incorporates the sea bottom prediction model and our original dataset to predict the potential area with the presence of fish in the sea. Feature selection extracts the necessary features to build our machine learning model. The output of this model corresponds to the resolution grid with fish abundance.

4. **Model Evaluation**

After training and testing of our spatial-temporal dataset (fisheries data and satellite data) our predictive model is evaluated in two phases. The bottom temperature forecasting regression model is evaluated through MAE and RMSE. For prediction of fish locations through gradient boost model (classification model) F1 score, Recall and ROC analysis are used.

Advantages of Concept 3

1. Inclusion of environmental data like sea bottom temperature to our fisheries data increases paves way for higher predictability of fish species in sea.
2. The correlation between fish abundance and sea temperature establishes an important relationship in decision making.

Disadvantages of Concept 3

1. Quality of fisheries data for example landing data for this model greatly affects the overall accuracy. Detected fault tolerances can arise from misreporting of landing information.
2. To ensure a more dependable and more accurate model, there is need to augment the overall architecture with monitoring techniques of fish species in sea.

3.8.5 First iteration

The matrix was created on the Excel platform. The study's author chose the instances and determined a certain selection criterion. The researcher performed priority rating and weighing displayed in the matrix. The outcome can be summarized below.

Table 3: Evaluation matrix iteration 1

	CRITERIA/CHALLENGES	PRIORITY	WEIGHT	CONCEPT 1	CONCEPT 2	CONCEPT 3
COST	CAPEX		5	4	2	2
	OPEX		3	4	2	4
		3	SUM	9.6	4.8	6.6
TECHNOLOGY	MATURITY		5	4	2	1
	TQP DURATION		3	4	2	1
	ALGORITHM COMPLEXITY	4	4	4	2	3
	TECHNICAL SAFETY		2	4	3	1
			SUM	11.2	6	4.4
OPERATION	SC INTEGRATION		5	2	3	1
	FLEXIBILITY FOR FUTURE USE		2	4	2	3
	INTEGRATION WITH MRP/ERP	3	3	2	3	1
	MANAGEMENT BOOST		4	3	5	1
	LIFT CAPACITY OF FISHING INSUDTRY		5	5	2	1
			SUM	7.3	7	2.8
	WEIGHTED AVERAGE:			**28.1**	**17.8**	**13.8**

CAPEX rating (Capital Expenditure) refers to the expenditure that our company of interest might incur in trying to refurbish or maintain the available physical assets in order to incorporate the predictive machine learning algorithm architecture.

OPEX rating (Operating Expenditure) refers to the expenditure in terms of operating costs, administrative costs, utility bills or maintenance expenses that our company of interest incurs daily in order to keep the predictive model running. Both ratings offer valuable insights to the integration of our proposed model.

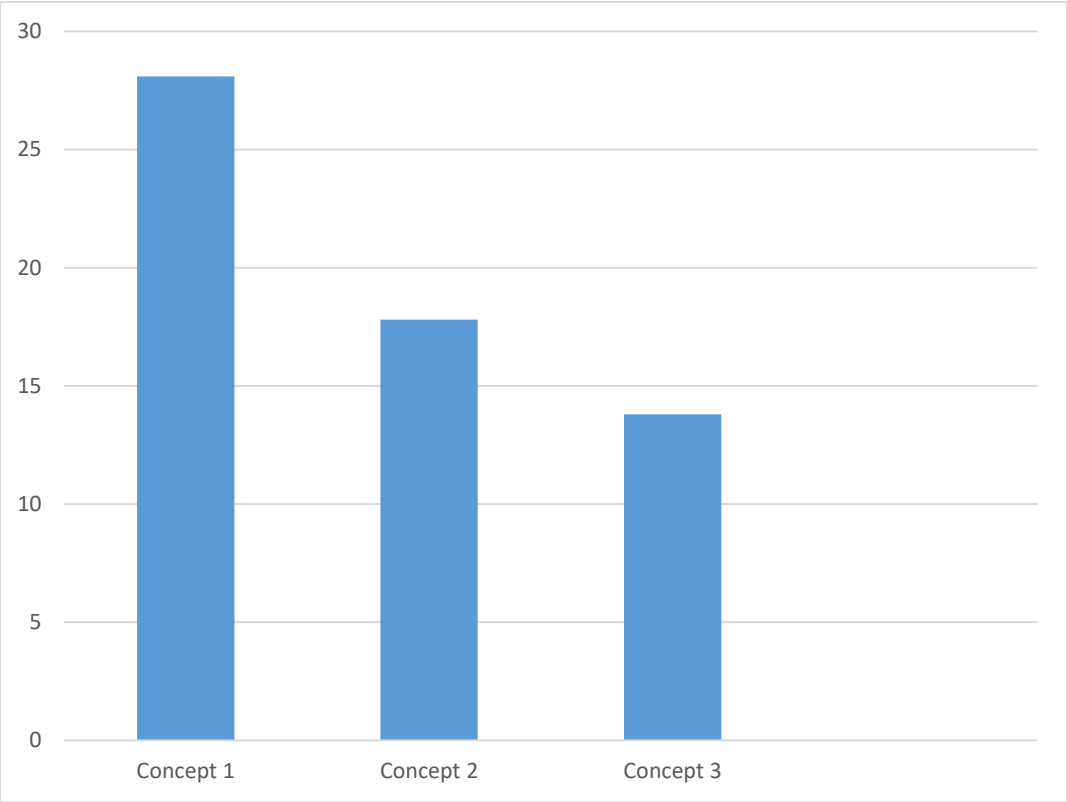


Figure 10: Graph showing scores distribution among concepts

On a Likert scale of 1 to 5, with 5 being the best, all weighting and scoring was done. The sums for each category in the matrix are determined using the formula below:
The category sum is calculated as follows:

$$Category\ sum\ = \frac{\sum(criteria\ score * weight)}{no\ of\ criteria\ within\ the\ category} * \frac{Priority}{no\ of\ categories}$$

3.8.6 Second iteration

At the conclusion of screening process, a bar chart was drawn. It is based on the category sums for each design option. The concept selection for our models was done using Pugh Matrix [79]. The bottom row of the matrix contains the weighted average (total sum). After removing extraneous notions, the below formulas were employed in the matrix below for the second iteration:

$$Criteria\ weighted\ score = \frac{criteria\ score}{4(\max score)} * weight[\%]$$

$$Subcategory\ weighted\ score\ = \sum(criteria\ \ weighted\ score\ [\%])$$

$$Category\ sum\ = \frac{subcategory\ weighted\ score[\%]}{100} * priority[\%]$$

Table 4: Evaluation matrix using Pugh Matrix Iteration 2

	Priority (%)	evaluation criteria	Weight (%)	concept 1		concept 2	
				scores	weighted	scores	weighted
COST	10	Hardware cost	65	2	22%	6	65%
		Life of field	35	4	23%	4	23%
		sub-category weighted score			45%		88%
		category sum		5%		2%	
MATURITY	20	Algorithm complexity	40	3	20%	5	33%
		Qualification effort	60	4	40%	6	60%
		sub-category weighted score			60%		93%
		category sum		12%		4%	
CONTROL AND OPERATION	20	Algorithm deployment	50	5	42%	6	50%
		Simplicity of control	50	5	42%	6	50%
		sub-category weighted score			60%		93%
		category sum		17%		11%	
INSTALLATION	10	Operability	20	2	7%	6	20%
		Flexibility for integration with future projects	25	6	25%	6	25%
		Rapport with other systems	40	3	20%	1	7%
				1	3%	6	15%
		sub-category weighted score			54%		67%
		category sum		27%		5%	
∑ (%): 100		Overall weighted score		61%		22%	

From the weighted decision matrix shown above, the chosen concept is the first concept which involves multiple regression algorithms to predict location of fish abundance in sea and as well as generating forecasts. Modelling of these machine learning models is done in Jupyter Notebook with python software with packages of these algorithms which involves mathematical modeling, training of data and a simulation platform.

3.8.7 Development of Chosen Concept

This stage mainly focuses on developing our chosen solutions in a more simplified mathematical modeling approach which breaks down the implementation to address our main objective in question.

Mathematical Modelling of Concepts

3.8.7.1 Model A: Linear Regression

This type of regression algorithm models a relationship between independent variables and dependent variables. For a dependent *y* variable is due to independent variable *x.* The major focus is modeling a relationship between *y* and *x* and forecasting new predictions.

This regression model can be represented by equation:

$$y = \beta_0 + \beta_1 + \varepsilon$$

Where β_0 is the regression line intercept at *x = 0,* β_1= gradient of the regression line (increasing or decreasing) and ε = residual value in model (negligible preferred).

This is to define that, to what extent do the independent variables (being measured on categorical values) influence dependent variables. This model can also be used to handle a large number of observations, by modeling a linear relationship between independent variables *x* and a single dependent variable *y* through multilinear regression analysis [80].

3.8.7.2 Model B: Decision Trees

A Decision Tree algorithm is suitable for both classification and regression tasks. Through continual usage of this algorithm, it has been discovered that the DT regressor does not perform up to standard on limited depth but participates dependently at greater depths hence associated with overfitting issues. Popularly known for having the leaf node and root node which splits into two nodes after reaching a threshold value [81].

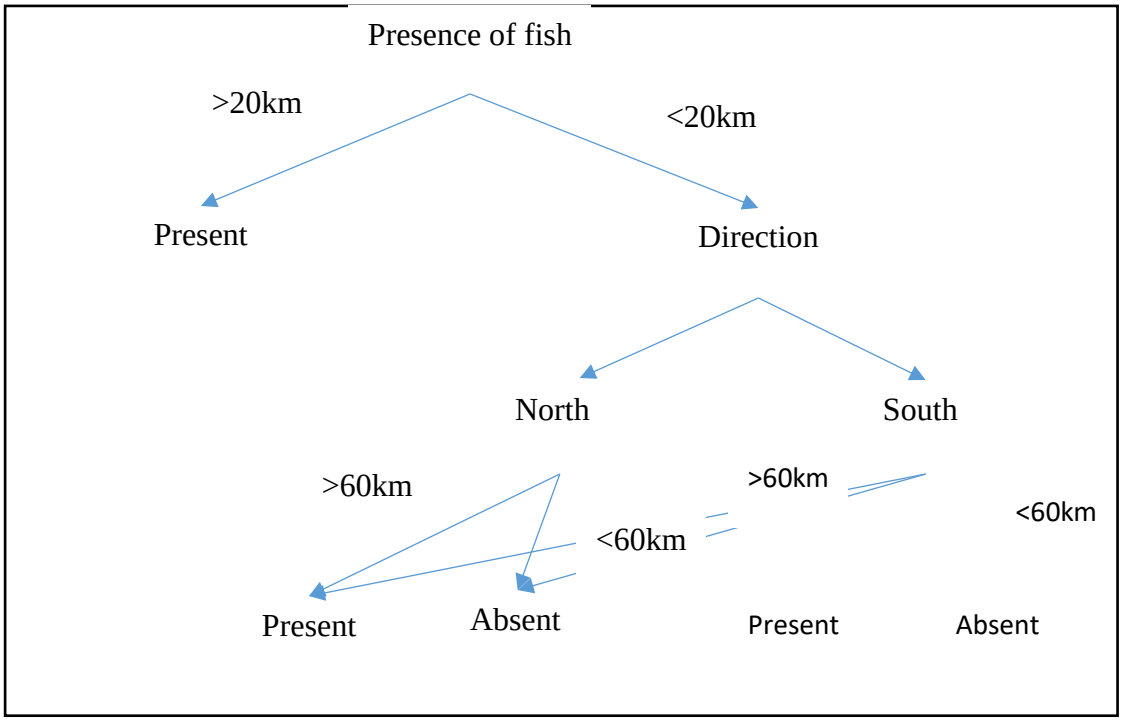


Figure 11: Decision Tree classifying fish location by distance and direction to determine feasibility of having a profitable catch

The basis of building a regression or a classification decision tree algorithm is on inbound and outbound pairs as shown in equation below:

$$(\left(x^i, y^i\right) \in (X * Y))1_{i=1}^{n}$$

For every available test node, a split that is considered to be good (S_r, S_l) amongst the sub-observations, is selected amongst *p* inbound attributes accompanied with an optimal cut off point. By selecting the best split amongst observations, average impurity is reduced.

$$\Delta I(\left(y^i\right)_{i\in s}, \left(y^i\right)_{i\in s_r}, (y^i)_{i\in s_l}$$

$I(\left(y^i\right)_{i\in s} = 0$ represents a cut off criterion where the is no impurity, *I* is the outbound impurity represented as variance in regression tasks and lack of predictability in classification tasks. Since algorithms come with different easiness in computational requirements, two splitting criterions are usually used the Gini index Impurity criterion for most classification problems and Information gain criterion.

3.8.7.3 Model D: Support Vector Machines (SVM)

Support Vector Machine (SVM) is a classifier in binary form that aims to separate training samples /class points by making use of hyperplanes. After finding the best hyperplane to sperate these points, then distance is calculated of each point in the sample towards the hyperplane. Three concepts are usually applied which are hard margin, soft, soft margin and kernel function in order to produce SVM and classifying data better. Kernel functions are employed when some data samples cannot be separated by any hyperplane to produce non-linear separating hyperplanes. Lagrange functions are used on eigen vector mapped to find correct hyperplane. The points lying on the boundaries from either set of samples but near to the hyperplane are called support vectors [82].

Increase in usability of SVM algorithms is its ability to have a high memory space and an effective approach towards high dimensional datasets. For data with clear separated hyperplanes, it become much more efficient. On another note, SVM on practical cases handling of large sample data sets takes time leading to a lower expected training efficiency [83].

For *n* data points, constituting of *x* training samples over a range of targets $y \in (-1, 1)$. For establishing a tradeoff between quality of separation of two sets and having bigger margin size, λ is used. The separating hyperplane has a normal vector , *w* and this parameter $\frac{b}{\|w\|}$ is responsible for offseting the hyperplane from origin [84].

3.8.7.4 Model E: Gradient Boost Regression (GBR)

Gradient Boosting Regression is an ensemble learning method, this approach banks on capitalizing on the combination of weak models to produce more stronger ensemble prediction. Whilst Random Forest uses the same approach but relying on averaging of models, GBR at each stage focuses on training of a new weak/base learner and it is added on the ensemble in a sequential manner. On the same note, this means the newly added base learner fits the available prediction error of the already existing collection of base learners [85].

One good advantage of GBR is its model's ability to reduce variance and bias. One of the drawbacks associated with it, is the overfitting of data when too many learners are involved [86] .

The principle behind GBR is to aim maximally, for base learners to suit with negative gradient of the loss function. This done to minimize consecutive error fitting associated with the use of any arbitrary loss function though selection of the loss function depends on the user

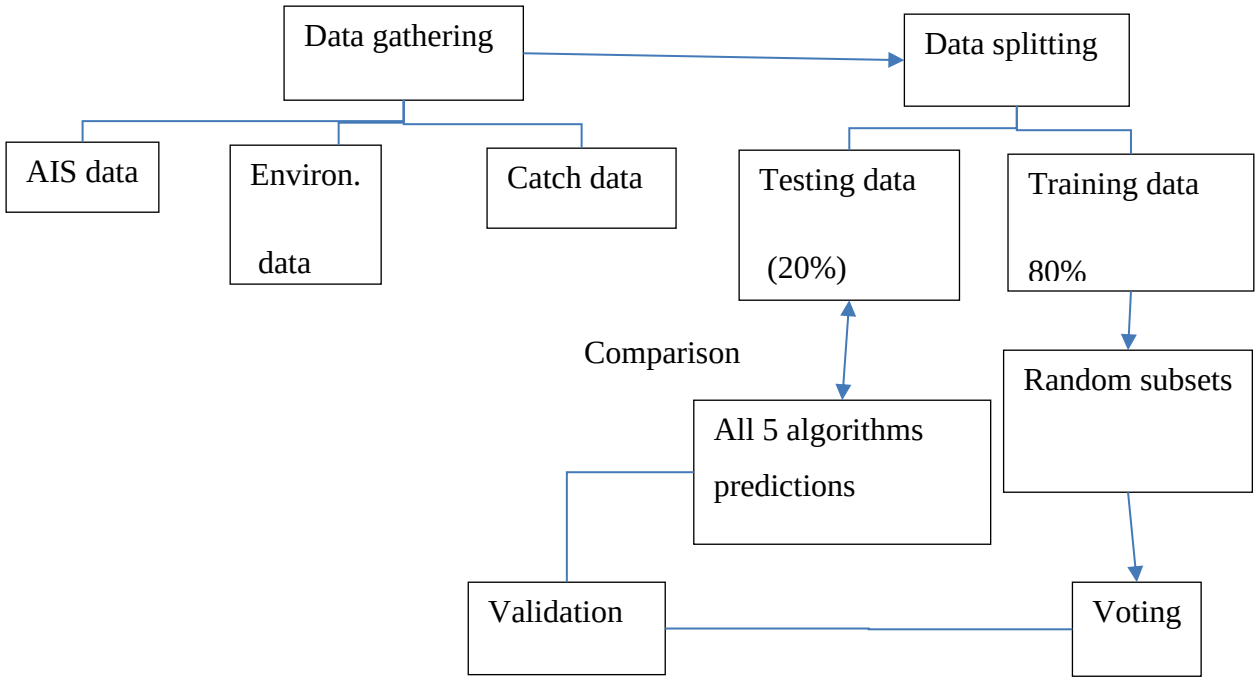
implementing his own specific task loss. With the easiness of GBR algorithm to implement, there is greater freedom when choosing the correct loss function and models design [87].

Gradient Boost algorithm

For a given specific loss function $\psi(y, f)$ and a base learner $h\ (x, \theta)$, a new function is represented as $h(x, \theta_t)$ is chosen arbitrarily which parallels the negative gradient the most $\{g_t(x_i)\}_{i=1}^{N}$ along our observation [87]:

$$g_t(x) = E_y \left[\frac{\delta \psi(y, f(x))}{\delta f(x)} \parallel x \right]_{f(x) = f^{t-1}(x)}$$

The diagram below shows the steps of the overall plan for training and implementation of our algorithms of choice in concept 1.



3.8.8 Data availability

Vessel data for Oddvar Nes, two fishing vessels (Trondskjær and Lise Beate) was extracted from Global Fishing Watch. This site is an open access online tool for visualization of vessel-based activities for a lot of commercial fishing vessels. Satellite monitoring and terrestrial receivers collect Automatic Identification System (AIS) data and integrated by vessel monitoring system data, environmental data and catch data to determine potential fishing effort with good catches.

The map allows for searching for vessel names through using Maritime Mobile Service Identity number (MMSI), International Maritime Organization number (IMO) and Vessel Monitoring System Identifier (VMSI). For our vessels of interest, Trondskjær and Lise Beate are denoted LFBC and LDQN respectively.

4 Chapter 4: Software Development, Methods and Results

This Chapter addresses important critical steps in the application of our chosen Concept 1, with five machine learning regression algorithms models: Support Vector Machines, Random Forest, Decision Trees, Gradient Boost Regression and Linear Regression. This section was done through Jupyter Notebook and PyCharm using Python 3.12. A detailed description of each subsection is provided developing from our problem statement, objectives of this study, data preparatory stages, training stage, tuning of our algorithms related parameters till we validate and evaluate performance measurement of algorithms in question and selection of the best model.

4.1 Objectives of implementing IDE

The application of Jupyter Notebook and PyCharm with Python 3.12 from an Integrated Development environment aims at meeting this project main research questions which can be summarized below:

1. To model, train and test predictive algorithms which can analyze multiple variables to predict location or abundance of fish in sea.
2. To establish correlation relationships between static and predictor variables in fisheries towards a higher catch effort.
3. To verify and validate the algorithm modelled in (a) above.
4. To select the best predictive algorithm(s) based on performance metrics.

4.2 Stages in Implementation

This summarizes our overall experimental plan in order to meet our research objectives, this can also be described as the modeling part of Objective 1. To be able to implement our machine learning algorithms whether its classification or regression tasks, data is collected, necessary features are extracted then a model is build based on the available training data and the model is evaluated, validated and pass through the accuracy test and finally approved for deployment. Regression algorithms fitted our problem of interest since we needed to discover relations between input variables and output to model a predictive futuristic behavior of our system.

4.2.1 Importing necessary libraries

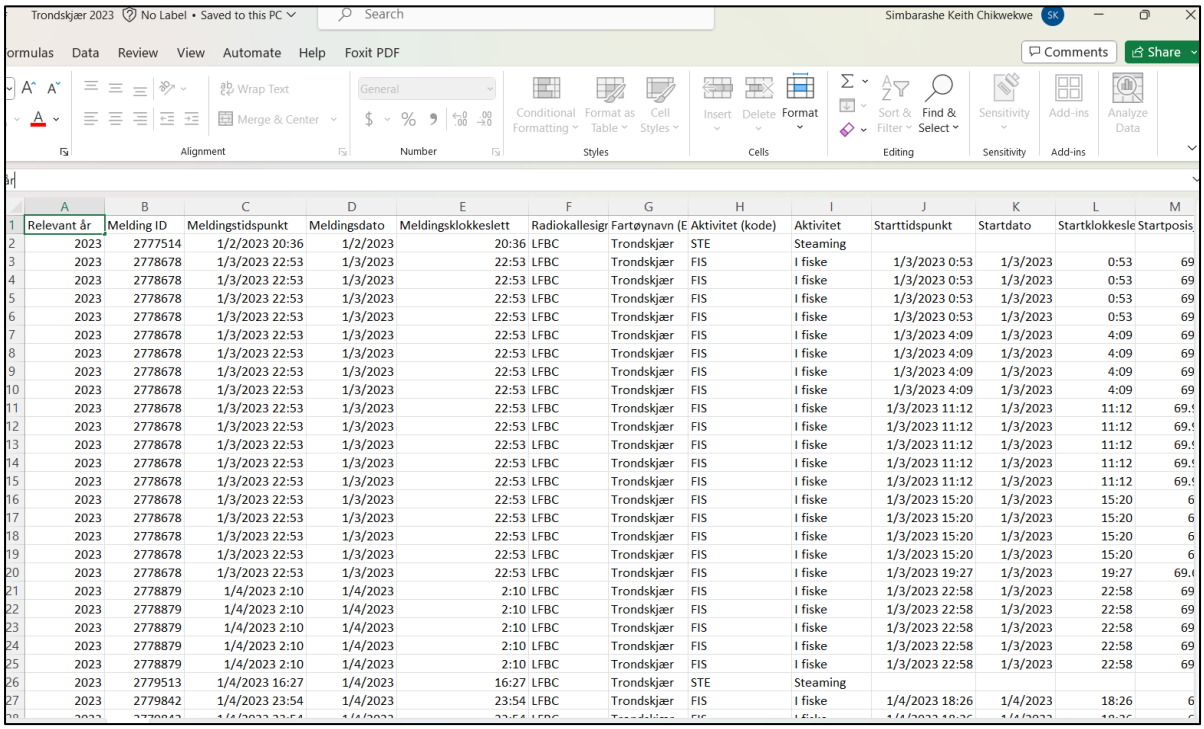
In addition to the Python 3.12 via Jupyter Notebook and PyCharm, multiple libraries and frameworks were called out to perform tasks of data collection, analysis and prediction tasks. Below is a summary of libraries and their specific functions that were used in this IDE:

- ***NumPy:*** This is a library in python responsible for handling numerical calculations through use of arrays and matrices.
- ***Pandas:*** This library is responsible for features that deals directly with the data frame inclusive of shaping and cleaning.
- ***Matplotlib:*** This library produces visuals or interactive plots in form of charts or graphs.
- ***SciPy.* Stats*:*** This module provides statistical tools for handling normality, asymmetrical patterns or hypothesis tests in datasets.
- ***Pickle:*** This module serves through loading predictive models as well serializing.
- **Sklearn.ensemble*:*** Two machine learning algorithms Random Forest Regressor and Gradient Boost Regressor were withdrawn, RFR combines multiple decision trees whilst GBR combine weaker models to create a strong predictive model through gradient boosting.
- ***Folium:*** This library produces maps in python to view geospatial data.
- ***Seaborn:*** This library manages statistical visuals in python and works hand in hand with matplotlib.
- ***Sklearn.impute:*** Simple imputer is imported from this library which is a tool responsible for handling missing values in a dataset through different strategies like mean imputation and mode imputation.
- ***Sklearn.preprocessing:*** Robert scaler is imported from this module which is a tool for handling outliers in a dataset through strategies like calculating interquartile range.
- ***Sklearn.metrics:*** Performance metrics confusion matrix, classification report and accuracy report were extracted from this module.
- ***Sklearn.model_selection:*** This module is extracted from the scikit-learn library and the train-test-split function is imported which is used to split a dataset into training and testing sets. This paves way for the evaluation of performance of a machine learning model on unseen data.
- ***Sklearn.tree:*** Decision Tree regressor algorithm was imported from this library.
- ***Sklearn.svm:*** The support vector machine regressor algorithm for regression tasks was imported.

- ***Sklearn.linear_model:*** Logistic Regression and Linear Regression responsible for handling classification and regression tasks are withdrawn.
- ***Sklearn.multioutput:*** A tool called multioutput regressor for handling multi-output regression problems. It fits a separate regression model for each output variable.
- ***Sklearn.metrics:*** The evaluation metrics MAE, MSE and R2 score were imported from this library. These metric functions are used to evaluate the performance of a regression model. Errors between actual and predicted observations can be calculated.

4.2.2 Datasets

These experiments made use of two datasets of the two fishing vessels at Oddvar Nes, Trondskjær and Lise Beate. AIS data in form longitude/latitude and catch data in form of catch weights, types of species etc. and other useful data like radio signals, notification times etc. was used. This data was for one fishing season, 2023. Below, Figure 12 is an extract of the dataset from Trondskjær in excel format, like wise Lise Beate dataset was in the same format.



	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Relevant år	Melding ID	Meldingstidspunkt	Meldingsdato	Meldingsklokkeslett	Radiokallesigr	Fartøynavn (E	Aktivitet (kode)	Aktivitet	Starttidspunkt	Startdato	Startklokkesle	Startposis
2	2023	2777514	1/2/2023 20:36	1/2/2023	20:36	LFBC	Trondskjær	STE	Steaming				
3	2023	2778678	1/3/2023 22:53	1/3/2023	22:53	LFBC	Trondskjær	FIS	I fiske	1/3/2023 0:53	1/3/2023	0:53	69
4	2023	2778678	1/3/2023 22:53	1/3/2023	22:53	LFBC	Trondskjær	FIS	I fiske	1/3/2023 0:53	1/3/2023	0:53	69
5	2023	2778678	1/3/2023 22:53	1/3/2023	22:53	LFBC	Trondskjær	FIS	I fiske	1/3/2023 0:53	1/3/2023	0:53	69
6	2023	2778678	1/3/2023 22:53	1/3/2023	22:53	LFBC	Trondskjær	FIS	I fiske	1/3/2023 0:53	1/3/2023	0:53	69
7	2023	2778678	1/3/2023 22:53	1/3/2023	22:53	LFBC	Trondskjær	FIS	I fiske	1/3/2023 4:09	1/3/2023	4:09	69
8	2023	2778678	1/3/2023 22:53	1/3/2023	22:53	LFBC	Trondskjær	FIS	I fiske	1/3/2023 4:09	1/3/2023	4:09	69
9	2023	2778678	1/3/2023 22:53	1/3/2023	22:53	LFBC	Trondskjær	FIS	I fiske	1/3/2023 4:09	1/3/2023	4:09	69
10	2023	2778678	1/3/2023 22:53	1/3/2023	22:53	LFBC	Trondskjær	FIS	I fiske	1/3/2023 4:09	1/3/2023	4:09	69
11	2023	2778678	1/3/2023 22:53	1/3/2023	22:53	LFBC	Trondskjær	FIS	I fiske	1/3/2023 11:12	1/3/2023	11:12	69.(
12	2023	2778678	1/3/2023 22:53	1/3/2023	22:53	LFBC	Trondskjær	FIS	I fiske	1/3/2023 11:12	1/3/2023	11:12	69.(
13	2023	2778678	1/3/2023 22:53	1/3/2023	22:53	LFBC	Trondskjær	FIS	I fiske	1/3/2023 11:12	1/3/2023	11:12	69.(
14	2023	2778678	1/3/2023 22:53	1/3/2023	22:53	LFBC	Trondskjær	FIS	I fiske	1/3/2023 11:12	1/3/2023	11:12	69.(
15	2023	2778678	1/3/2023 22:53	1/3/2023	22:53	LFBC	Trondskjær	FIS	I fiske	1/3/2023 11:12	1/3/2023	11:12	69.(
16	2023	2778678	1/3/2023 22:53	1/3/2023	22:53	LFBC	Trondskjær	FIS	I fiske	1/3/2023 15:20	1/3/2023	15:20	6
17	2023	2778678	1/3/2023 22:53	1/3/2023	22:53	LFBC	Trondskjær	FIS	I fiske	1/3/2023 15:20	1/3/2023	15:20	6
18	2023	2778678	1/3/2023 22:53	1/3/2023	22:53	LFBC	Trondskjær	FIS	I fiske	1/3/2023 15:20	1/3/2023	15:20	6
19	2023	2778678	1/3/2023 22:53	1/3/2023	22:53	LFBC	Trondskjær	FIS	I fiske	1/3/2023 15:20	1/3/2023	15:20	6
20	2023	2778678	1/3/2023 22:53	1/3/2023	22:53	LFBC	Trondskjær	FIS	I fiske	1/3/2023 19:27	1/3/2023	19:27	69.(
21	2023	2778879	1/4/2023 2:10	1/4/2023	2:10	LFBC	Trondskjær	FIS	I fiske	1/3/2023 22:58	1/3/2023	22:58	69
22	2023	2778879	1/4/2023 2:10	1/4/2023	2:10	LFBC	Trondskjær	FIS	I fiske	1/3/2023 22:58	1/3/2023	22:58	69
23	2023	2778879	1/4/2023 2:10	1/4/2023	2:10	LFBC	Trondskjær	FIS	I fiske	1/3/2023 22:58	1/3/2023	22:58	69
24	2023	2778879	1/4/2023 2:10	1/4/2023	2:10	LFBC	Trondskjær	FIS	I fiske	1/3/2023 22:58	1/3/2023	22:58	69
25	2023	2778879	1/4/2023 2:10	1/4/2023	2:10	LFBC	Trondskjær	FIS	I fiske	1/3/2023 22:58	1/3/2023	22:58	69
26	2023	2779513	1/4/2023 16:27	1/4/2023	16:27	LFBC	Trondskjær	STE	Steaming				
27	2023	2779842	1/4/2023 23:54	1/4/2023	23:54	LFBC	Trondskjær	FIS	I fiske	1/4/2023 18:26	1/4/2023	18:26	6

Figure 12: The dataset of Trondskjær fishing vessel consisting of AIS data and catch data for the season 2023

The available datasets “Trondskjær 2023” and “Lise Beate 2023” in excel formats are loaded through pandas to enable data manipulation and analysis. This utilizes a structure called data frame. The columns are printed with each variable name attached whilst rows represent observations. Figure 13 below shows our columns as shown in our dataset.

```
# Load the first Excel file "Lise Beate 2023"
lise_beate_df = pd.read_excel("Lise Beate 2023.xlsx")

# Print the column names of the "Lise Beate 2023" DataFrame
print("Column names of the 'Lise Beate 2023' DataFrame:")
print(lise_beate_df.columns.tolist())

# Load the second Excel file "Trondskjær 2023"
trondskjaer_df = pd.read_excel("Trondskjær 2023.xlsx")

# Print the column names of the "Trondskjær 2023" DataFrame
print("\nColumn names of the 'Trondskjær 2023' DataFrame:")
print(trondskjaer_df.columns.tolist())
```

```
Column names of the 'Lise Beate 2023' DataFrame:
['Relevant år', 'Melding ID', 'Meldingstidspunkt', 'Meldingsdato', 'Meldingsklokkeslett', 'Radiokallesignal (ERS)', 'Fartøynavn (ERS)', 'Aktivitet (kod
e)', 'Aktivitet', 'Starttidspunkt', 'Startdato', 'Startklokkeslett', 'Startposisjon bredde', 'Startposisjon lengde', 'Lokasjon start (kode)', 'Havdybde s
tart', 'Stopptidspunkt', 'Stoppdato', 'Stoppklokkeslett', 'Varighet', 'Stopposisjon bredde', 'Stopposisjon lengde', 'Lokasjon stopp (kode)', 'Områdegrupp
ering stopp (kode)', 'Havdybde stopp', 'Trekkavstand', 'Redskap FAO', 'Hovedart - FDIR (kode)', 'Art FAO (kode)', 'Art FAO', 'Art - hovedgruppe (kode)',
'Art - hovedgruppe', 'Rundvekt']

Column names of the 'Trondskjær 2023' DataFrame:
['Relevant år', 'Melding ID', 'Meldingstidspunkt', 'Meldingsdato', 'Meldingsklokkeslett', 'Radiokallesignal (ERS)', 'Fartøynavn (ERS)', 'Aktivitet (kod
e)', 'Aktivitet', 'Starttidspunkt', 'Startdato', 'Startklokkeslett', 'Startposisjon bredde', 'Startposisjon lengde', 'Lokasjon start (kode)', 'Havdybde s
tart', 'Stopptidspunkt', 'Stoppdato', 'Stoppklokkeslett', 'Varighet', 'Stopposisjon bredde', 'Stopposisjon lengde', 'Lokasjon stopp (kode)', 'Områdegrupp
ering stopp (kode)', 'Trekkavstand', 'Redskap FAO', 'Hovedart - FDIR (kode)', 'Art FAO (kode)', 'Art FAO', 'Art - hovedgruppe (kode)', 'Art - hovedgrupp
e', 'Rundvekt']
```

Figure 13: Loading of two data frames of interest by pandas

After loading the datasets, the shape of our datasets is determined. Trondskjær 2023 had 960 samples and 32 features. Lise Beate had 887 samples and 33 features. The available dataset has 960 and 887 observations available for training or testing respectively. The dimensionality of these data-frames indicates 32 and 33 attributes respectively. The need to know the shape of our dataset is a critical step before data preprocessing.

```
df1= trondskjaer_df
df1.shape
```

```
(960, 32)
```

```
df2= lise_beate_df
df2.shape
```

```
(887, 33)
```

Figure 14: A snippet of observations and features from each dataset

4.2.2.1 Intersection and merging of datasets

On this stage, the two datasets are compared using column names from the loaded dataset to get the common columns between the datasets. After that, the columns that are not common between the datasets are dropped. Finally, the two datasets are merged by using common values in the specified columns. Rows that consist of matching values are included in the merged dataset resulting in a merged data-frame.

```
# Get the common columns between the datasets
common_cols = df1.columns.intersection(df2.columns)

# Drop the columns that are not common between the datasets
df1_common = df1[common_cols]  # select only the common columns from df1
df2_common = df2[common_cols]  # select only the common columns from df2

# Merge the datasets on the common columns
merged_df = pd.concat([df1_common, df2_common], ignore_index=True)  # concatenate the datasets along the row axis

# Print the resulting dataset
merged_df.head()
```

	Relevant år	Melding ID	Meldingstidspunkt	Meldingsdato	Meldingsklokkeslett	Radiokallesignal (ERS)	Fartøynavn (ERS)	Aktivitet (kode)	Aktivitet	Starttidspunkt	...	Lokasjon stopp (kode)	Områdeg st
0	2023	2777514	2023-01-02 20:36:00	2023-01-02	20:36:00	LFBC	Trondskjær	STE	Steaming	NaT	...	NaN	
1	2023	2778678	2023-01-03 22:53:00	2023-01-03	22:53:00	LFBC	Trondskjær	FIS	I fiske	2023-01-03 00:53:00	...	29.0	
2	2023	2778678	2023-01-03 22:53:00	2023-01-03	22:53:00	LFBC	Trondskjær	FIS	I fiske	2023-01-03 00:53:00	...	29.0	
3	2023	2778678	2023-01-03 22:53:00	2023-01-03	22:53:00	LFBC	Trondskjær	FIS	I fiske	2023-01-03 00:53:00	...	29.0	
4	2023	2778678	2023-01-03 22:53:00	2023-01-03	22:53:00	LFBC	Trondskjær	FIS	I fiske	2023-01-03 00:53:00	...	29.0	

5 rows × 32 columns

Figure 15: Intersection and merging of dataset

The resultant merged dataset has 1847 samples and 32 features. A dictionary to map Norwegian named variables to their English names as shown in Figure 16 below.

```python
# Create a dictionary to map the Norwegian variable names to their English translations
translations = {
    'Relevant år': 'Relevant year',
    'Melding ID': 'Message ID',
    'Meldingstidspunkt': 'Message time',
    'Meldingsdato': 'Message date',
    'Meldingsklokkeslett': 'Message clock time',
    'Radiokallesignal (ERS)': 'Radio call signal (ERS)',
    'Fartøynavn (ERS)': 'Vessel name (ERS)',
    'Aktivitet (kode)': 'Activity (code)',
    'Aktivitet': 'Activity',
    'Starttidspunkt': 'Start time',
    'Startdato': 'Start date',
    'Startklokkeslett': 'Start clock time',
    'Startposisjon bredde': 'Start position latitude',
    'Startposisjon lengde': 'Start position longitude',
    'Lokasjon start (kode)': 'Location start (code)',
    'Havdybde start': 'Sea depth start',
    'Stopptidspunkt': 'Stop time',
    'Stoppdato': 'Stop date',
    'Stoppklokkeslett': 'Stop clock time',
    'Varighet': 'Duration',
    'Stopposisjon bredde': 'Stop position latitude',
    'Stopposisjon lengde': 'Stop position longitude',
    'Lokasjon stopp (kode)': 'Location stop (code)',
    'Områdegruppering stopp (kode)': 'Area group stop (code)',
    'Trekkavstand': 'Distance travelled',
    'Redskap FAO': 'FAO gear',
    'Hovedart - FDIR (kode)': 'Main species - FDIR (code)',
    'Art FAO (kode)': 'Species FAO (code)',
    'Art FAO': 'Species FAO',
    'Art - hovedgruppe (kode)': 'Species - main group (code)',
    'Art - hovedgruppe': 'Species - main group',
    'Rundvekt': 'Round weight'
}
```

Figure 16: Norwegian variables translated to English names

The resultant final merged dataset (1847 samples, 32 features) with their variable names translated to English are shown in Figure 17 below.

	Relevant year	Message ID	Message time	Message date	Message clock time	Radio call signal (ERS)	Vessel name (ERS)	Activity (code)	Activity	Start time	...	Location stop (code)	Area group stop (code)	Distance travelled	FAO gear	Main species - FDIR (code)	Sl (
0	2023	2777514	2023-01-02 20:36:00	2023-01-02	20:36:00	LFBC	Trondskjær	STE	Steaming	NaT	...	NaN	NaN	NaN	NaN	NaN	
1	2023	2778678	2023-01-03 22:53:00	2023-01-03	22:53:00	LFBC	Trondskjær	FIS	I fiske	2023-01-03 00:53:00	...	29.0	27_2_A_2	2565.0	Snurpenot/ringnot, et fartøy	1032.0	
2	2023	2778678	2023-01-03 22:53:00	2023-01-03	22:53:00	LFBC	Trondskjær	FIS	I fiske	2023-01-03 00:53:00	...	29.0	27_2_A_2	2565.0	Snurpenot/ringnot, et fartøy	1032.0	
3	2023	2778678	2023-01-03 22:53:00	2023-01-03	22:53:00	LFBC	Trondskjær	FIS	I fiske	2023-01-03 00:53:00	...	29.0	27_2_A_2	2565.0	Snurpenot/ringnot, et fartøy	1032.0	
4	2023	2778678	2023-01-03 22:53:00	2023-01-03	22:53:00	LFBC	Trondskjær	FIS	I fiske	2023-01-03 00:53:00	...	29.0	27_2_A_2	2565.0	Snurpenot/ringnot, et fartøy	1032.0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
1842	2023	3095738	2023-11-12 18:28:00	2023-11-12	18:28:00	LDQN	LISE BEATE	FIS	I fiske	2023-11-12 14:59:00	...	NaN	NaN	343.0	Snurpenot/ringnot, et fartøy	611.0	
1843	2023	3095738	2023-11-12 18:28:00	2023-11-12	18:28:00	LDQN	LISE BEATE	FIS	I fiske	2023-11-12 18:27:00	...	NaN	NaN	0.0	Snurpenot/ringnot, et fartøy	611.0	
1844	2023	3095738	2023-11-12 18:28:00	2023-11-12	18:28:00	LDQN	LISE BEATE	FIS	I fiske	2023-11-12 18:27:00	...	NaN	NaN	0.0	Snurpenot/ringnot, et fartøy	611.0	
1845	2023	3095738	2023-11-12 18:28:00	2023-11-12	18:28:00	LDQN	LISE BEATE	FIS	I fiske	2023-11-12 18:27:00	...	NaN	NaN	0.0	Snurpenot/ringnot, et fartøy	611.0	
1846	2023	3096355	2023-11-13 06:44:00	2023-11-13	06:44:00	LDQN	LISE BEATE	STE	Steaming	NaT	...	NaN	NaN	NaN	NaN	NaN	

Figure 17: Resultant merged dataset with English headings

4.2.3 Data Cleaning and Preprocessing

Data cleaning and preprocessing procedure is an essential preliminary step in our analysis, laying the foundation for deep exploratory and inferential analysis. By addressing issues such as missing values, data types, duplicates and outliers we ensure the reliability, integrity and usability of our data for extracting informed decisions. Below is a description and explanation of how each step contributes to our analysis.

1. **Handle missing values-** The integrity of our outcome is greatly influenced by how we handle and identify missing values in our dataset. So, it is crucial that this process is handled effectively for reliable results. Different techniques are used for computing or make up for missing values which are mean or mode imputation estimations, totally deleting rows and columns with missing values. Interpolating is another great method. After that, we are able to find underlying patterns and trends in our dataset.

The program identified missing values in variables starting form start time to round weight.

```
Missing values per column:
Relevant year                    0
Message ID                       0
Message time                     0
Message date                     0
Message clock time               0
Radio call signal (ERS)          0
Vessel name (ERS)                0
Activity (code)                  0
Activity                         0
Start time                     145
Start date                     145
Start clock time               145
Start position latitude        145
Start position longitude       145
Location start (code)          161
Sea depth start                145
Stop time                      145
Stop date                      145
Stop clock time                145
Duration                       145
Stop position latitude         145
Stop position longitude        145
Location stop (code)           155
Area group stop (code)         161
Distance travelled             145
FAO gear                       145
Main species - FDIR (code)     242
Species FAO (code)             242
Species FAO                    242
Species - main group (code)    242
Species - main group           242
Round weight                   242
dtype: int64
```

Figure 18: Missing values before handling them

The outcome of this step is summarized by Figure 19 below showing missing values per column. The start time/date and end time/date are a variable which we cannot quantify. Eventually it was dropped.

```
Missing values per column:
Relevant year                    0
Message ID                       0
Message time                     0
Message date                     0
Message clock time               0
Radio call signal (ERS)          0
Vessel name (ERS)                0
Activity (code)                  0
Activity                         0
Start time                     145
Start date                     145
Start clock time                 0
Start position latitude          0
Start position longitude         0
Location start (code)            0
Sea depth start                  0
Stop time                      145
Stop date                      145
Stop clock time                  0
Duration                         0
Stop position latitude           0
Stop position longitude          0
Location stop (code)             0
Area group stop (code)           0
Distance travelled               0
FAO gear                         0
Main species - FDIR (code)       0
Species FAO (code)               0
Species FAO                      0
Species - main group (code)      0
Species - main group             0
Round weight                     0
dtype: int64
```

Figure 19: A representation of missing values per column per each variable

2. **Data types-** Since we have columns with different data variables, it is important to group the variables into their classes of belonging. For data with numerical nature (integers or floats) it is classified under numerical variables. For data with strings or categorical in nature) it is classified as categorical variables. The following processes of data manipulation and data visualization are made easy. Date time in our dataset is parsed and formatted correctly. Species types, radio call signals, vessel names among other variable can be termed objects/ categorical variables. The below Figure 20 summarizes each variable data type per column.

```
Data types per column:
Relevant year                          float64
Message ID                             float64
Message time                    datetime64[ns]
Message date                    datetime64[ns]
Message clock time                     object
Radio call signal (ERS)                object
Vessel name (ERS)                      object
Activity (code)                        object
Activity                               object
Start time                      datetime64[ns]
Start date                      datetime64[ns]
Start clock time                       object
Start position latitude               float64
Start position longitude              float64
Location start (code)                 float64
Sea depth start                       float64
Stop time                       datetime64[ns]
Stop date                       datetime64[ns]
Stop clock time                        object
Duration                              float64
Stop position latitude                float64
Stop position longitude               float64
Location stop (code)                  float64
Area group stop (code)                 object
Distance travelled                    float64
FAO gear                               object
Main species - FDIR (code)            float64
Species FAO (code)                     object
Species FAO                            object
Species - main group (code)           float64
Species - main group                   object
Round weight                          float64
dtype: object
```

Figure 20: The various data types per each column

3. **Remove duplicates-** This stage tackles redundancy issues in dataset were existence of duplicate rows lowers the quality of our data. It is possible that duplicate rows arise as a result of flawed data collection systems or malicious attacks in systems. The need to eliminate duplicates prevents biased analysis in summary statistics. Our data had zero duplicate rows.
4. **Outliers-** This stage identifies data points that behave differently from the rest of the dataset. The need to deal with these issues is to avoid disturbances in summary statistics that can be negative influencing to the model's performance. Different techniques can be used to handle outliers inclusive of substituting extreme values with less extreme ones (winsorization). The analysis then become reflective of natural patterns or trends within data frame. The figure 20 below shows the outliers per numerical column.

```
Missing values per column:
Relevant year                    0
Message ID                       0
Message time                     0
Message date                     0
Message clock time               0
Radio call signal (ERS)          0
Vessel name (ERS)                0
Activity (code)                  0
Activity                         0
Start time                     145
Start date                     145
Start clock time               145
Start position latitude        145
Start position longitude       145
Location start (code)          161
Sea depth start                145
Stop time                      145
Stop date                      145
Stop clock time                145
Duration                       145
Stop position latitude         145
Stop position longitude        145
Location stop (code)           155
Area group stop (code)         161
Distance travelled             145
FAO gear                       145
Main species - FDIR (code)     242
Species FAO (code)             242
Species FAO                    242
Species - main group (code)    242
Species - main group           242
Round weight                   242
dtype: int64
```

Figure 21: A representation of outliers per each column

Handling of outliers was performed by Robust Scaling(scikit-learn) by fitting on the numerical data and transforming it to handle outliers. The Robust Scaler handles this operation by calculating the median and Interquartile Range (IQR) on each feature in our dataset. Figure 22 shows a representation of our data variables showing outliers.

```
Outliers per numerical column:
Relevant year                    0
Message ID                       0
Start position latitude         16
Start position longitude        21
Location start (code)            0
Sea depth start                 50
Duration                        23
Stop position latitude          16
Stop position longitude         20
Location stop (code)             0
Distance travelled               7
Main species - FDIR (code)      36
Species - main group (code)      0
Round weight                    37
dtype: int64
```

Figure 22: A representation of outliers in our dataset

After that, the Robust Scaler scales each feature by lessening the median and dividing by the IQR. Finally, we replace our numerical columns in the original data frame with the scaled data. The figure 22 below shows the program that handled the outliers and the first five rows of our data frame after the process.

```
# Handle outliers: Use the RobustScaler to handle outliers in numerical data
# Initialize the RobustScaler
robust_scaler = RobustScaler()

# Fit the RobustScaler on the numerical data and transform it to handle outliers
scaled_data = robust_scaler.fit_transform(data_df[numerical_cols])

# Replace the numerical columns in the original DataFrame with the scaled data
data_df[numerical_cols] = scaled_data

# Print the first 5 rows of the DataFrame after handling outliers
print("First 5 rows of the DataFrame after handling outliers:")
data_df.head()
```

```
First 5 rows of the DataFrame after handling outliers:
```

	Relevant year	Message ID	Message time	Message date	Message clock time	Radio call signal (ERS)	Vessel name (ERS)	Activity (code)	Activity	Start time	...	Location stop (code)	Area group stop (code)	Distance travelled	FAO gear	Main species - FDIR (code)	Sp (c
0	0.0	-0.373107	2023-01-02 20:36:00	2023-01-02	20:36:00	LFBC	Trondskjær	STE	Steaming	NaT	...	-0.817061	27_2_A_2	0.984124	Snurpenot/ringnot, et fartøy	0.540249	
1	0.0	-0.363072	2023-01-03 22:53:00	2023-01-03	22:53:00	LFBC	Trondskjær	FIS	I fiske	2023-01-03 00:53:00	...	0.000000	27_2_A_2	0.027932	Snurpenot/ringnot, et fartøy	0.000000	
2	0.0	-0.363072	2023-01-03 22:53:00	2023-01-03	22:53:00	LFBC	Trondskjær	FIS	I fiske	2023-01-03 00:53:00	...	0.000000	27_2_A_2	0.027932	Snurpenot/ringnot, et fartøy	0.000000	
3	0.0	-0.363072	2023-01-03 22:53:00	2023-01-03	22:53:00	LFBC	Trondskjær	FIS	I fiske	2023-01-03 00:53:00	...	0.000000	27_2_A_2	0.027932	Snurpenot/ringnot, et fartøy	0.000000	
4	0.0	-0.363072	2023-01-03 22:53:00	2023-01-03	22:53:00	LFBC	Trondskjær	FIS	I fiske	2023-01-03 00:53:00	...	0.000000	27_2_A_2	0.027932	Snurpenot/ringnot, et fartøy	0.000000	

5 rows × 32 columns

Figure 23: A program to handle outliers by Robust Scaler

The Simple imputer (scikit-learn) was used to handle missing values for both our categorical and numerical data columns. In this case, the strategy was implemented by mean imputation whereby Nan values were computed to mean of the column for the numerical columns and mode imputation for the categorical columns by replacing with the most frequent value in the column. The program below in figure 23 summarizes the above process for identifying missing values, outliers and duplicates in data cleaning and preprocessing.

```
# Handle missing values: Identify and handle missing data appropriately (e.g., imputation, deletion)
# Identify missing values
missing_values = english_df.isnull().sum()
print("Missing values per column:")
print(missing_values)

# Impute missing values using mean imputation for numerical columns and mode imputation for categorical columns
numerical_cols = english_df.select_dtypes(include=['int64', 'float64']).columns
categorical_cols = english_df.select_dtypes(include=['object']).columns

# Mean imputation for numerical columns
numerical_imputer = SimpleImputer(strategy='mean')
english_df[numerical_cols] = numerical_imputer.fit_transform(english_df[numerical_cols])

# Mode imputation for categorical columns
categorical_imputer = SimpleImputer(strategy='most_frequent')
english_df[categorical_cols] = categorical_imputer.fit_transform(english_df[categorical_cols])

# Data types: Ensure that data types are appropriate for each column
# Check data types
print("\nData types per column:")
print(english_df.dtypes)


# Remove duplicates: Check for and remove any duplicate rows
# Remove duplicates
data_df = english_df.drop_duplicates()
print("\nNumber of duplicate rows removed:", len(english_df) - len(data_df))

# Outliers: Identify and handle outliers if necessary
# Set a threshold for identifying outliers (e.g., 3 standard deviations from the mean)
threshold = 3

# Identify outliers for numerical columns
outliers = data_df[numerical_cols].apply(lambda x: (x > x.mean() + threshold * x.std()) | (x < x.mean() - threshold * x.std()))
print("\nOutliers per numerical column:")
print(outliers.sum())
```

Figure 24: A program for identifying outliers, duplicates and missing values

4.2.4 Descriptive Statistics

This is accomplished by exploring our data (Exploratory Data Analysis) to identify the true trends and relationships present. This process paves way for feature selection, hypothesis tests and more data manipulation strategies. To obtain this, we will undertake the following key steps:

1. **Summary Statistics:** Measures of central tendencies and dispersion like mean, mode, *nth* percentiles and standard variations are computed to gain and oversight of data distribution and variability. Lingering outliers can be identified or unusual trends. In this case, a few numerical variables are selected to represent the summary statistics.

	Start position longitude	Location start (code)	Sea depth start	\
count	1847.000000	1847.000000	1847.000000	
mean	-1.318700	-0.814542	-0.149935	
min	-16.336259	-5.192828	-6.688525	
25%	-0.800924	-0.814542	-0.393443	
50%	0.000000	0.000000	0.000000	
75%	0.199076	0.185458	0.606557	
max	8.605081	4.636454	3.286885	
std	4.740416	1.929578	1.228124	

	Stop time	Stop date	\
count	1702	1702	
mean	2023-03-10 11:55:18.225616896	2023-03-09 23:47:18.542890496	
min	2023-01-03 02:25:00	2023-01-03 00:00:00	
25%	2023-01-19 06:34:30	2023-01-19 00:00:00	
50%	2023-02-14 08:53:00	2023-02-14 00:00:00	
75%	2023-04-11 07:34:00	2023-04-11 00:00:00	
max	2023-11-12 18:27:00	2023-11-12 00:00:00	
std	NaN	NaN	

	Duration	Stop position latitude	Stop position longitude	\
count	1.847000e+03	1847.000000	1847.000000	
mean	1.077163e-16	-2.462895	-1.349507	
min	-2.079478e+00	-21.839458	-16.577861	
25%	-5.378109e-01	-0.789168	-0.818011	
50%	0.000000e+00	0.000000	0.000000	
75%	4.621891e-01	0.210832	0.181989	
max	6.149689e+00	2.810445	8.893996	
std	9.784750e-01	5.795660	4.824967	

	Location stop (code)	Distance travelled	Main species - FDIR (code)	\
count	1847.000000	1847.000000	1847.000000	
mean	-0.817061	0.984124	0.540249	
min	-4.939345	-0.408924	-42.100000	
25%	-0.817061	-0.256578	-1.000000	
50%	0.000000	0.000000	0.000000	
75%	0.182939	0.743422	0.000000	
max	4.390529	169.050158	117.000000	
std	1.921968	7.304392	16.091725	

Figure 25: A representation of summary statistics of numerical variables

2. **Distribution Visualization:** Our data frame is represented on histograms and box plots to gain knowledge of our data is spread and shaped. Through inspection of these visuals patterns and skewness present helps us identify how much deviations form normality and range the data exhibits. A few selected variables are represented on histogram and box plot below.

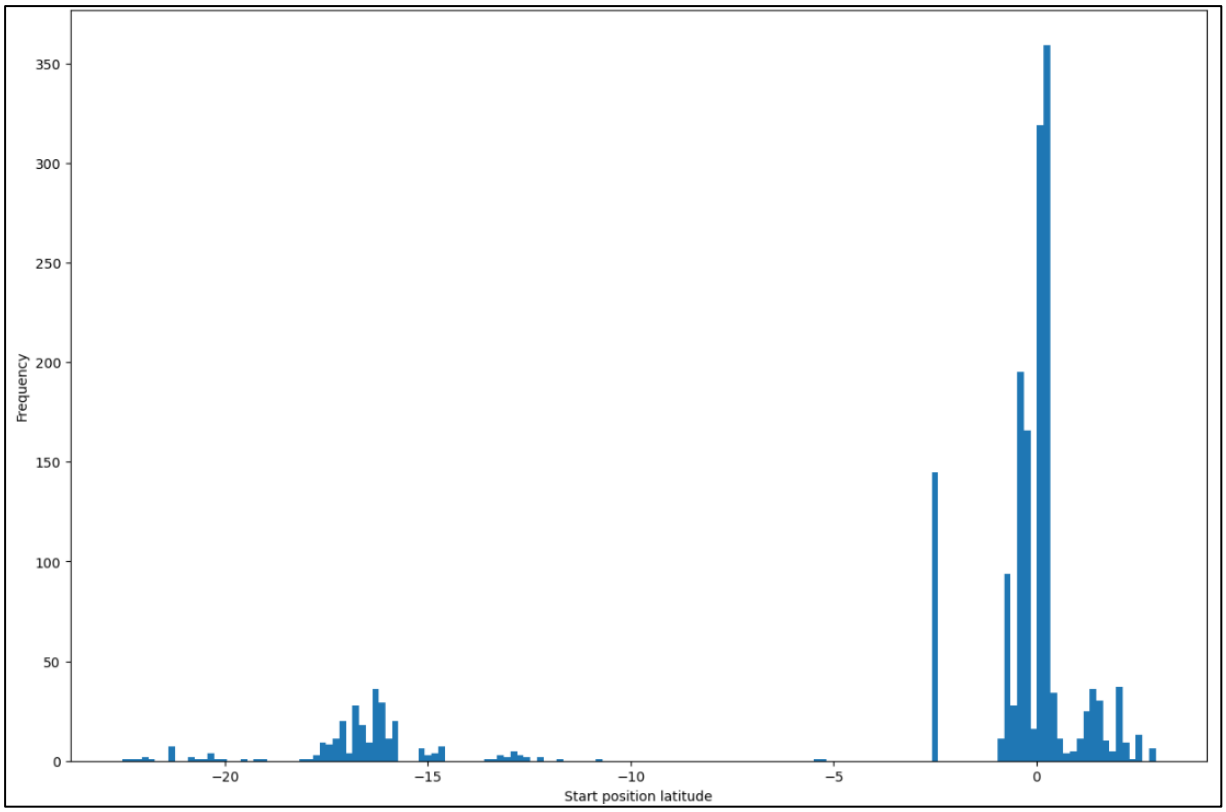


Figure 26: A histogram of start position latitude

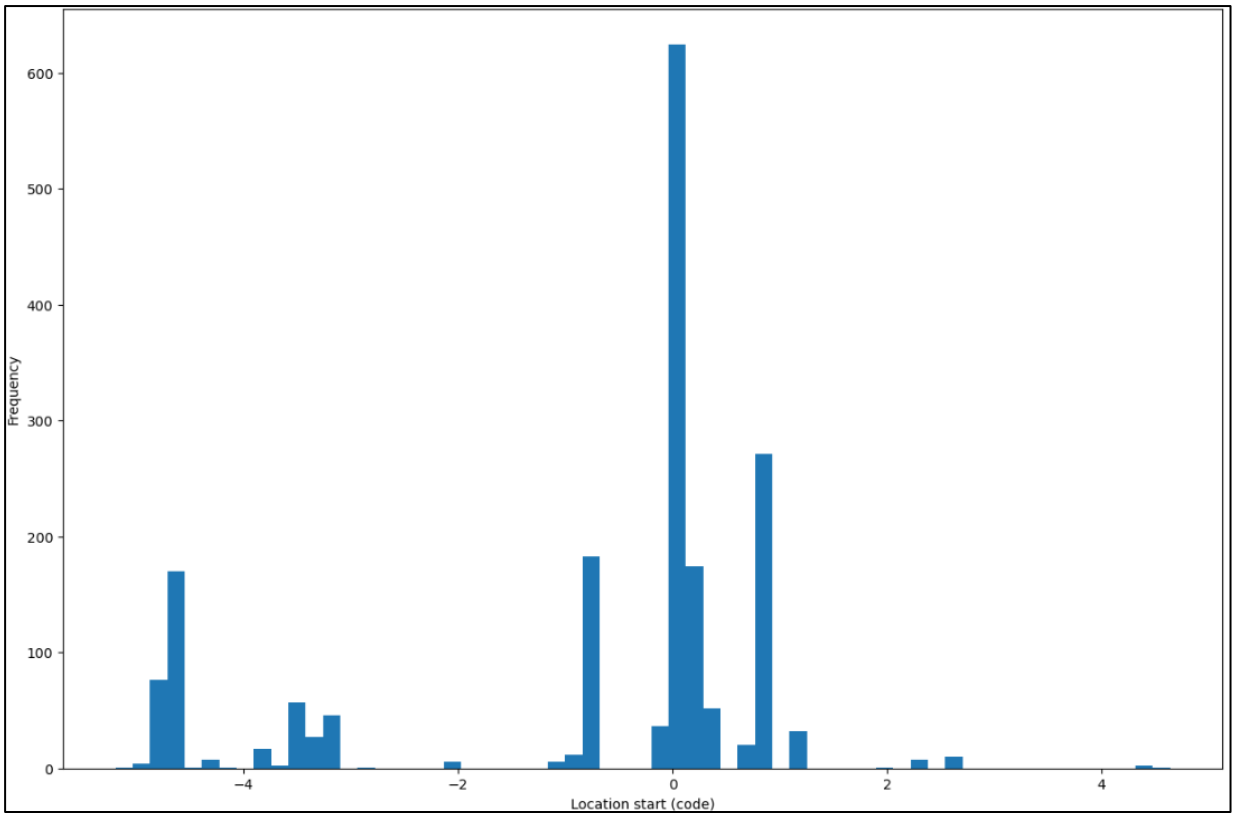


Figure 27: A histogram of location start (code)

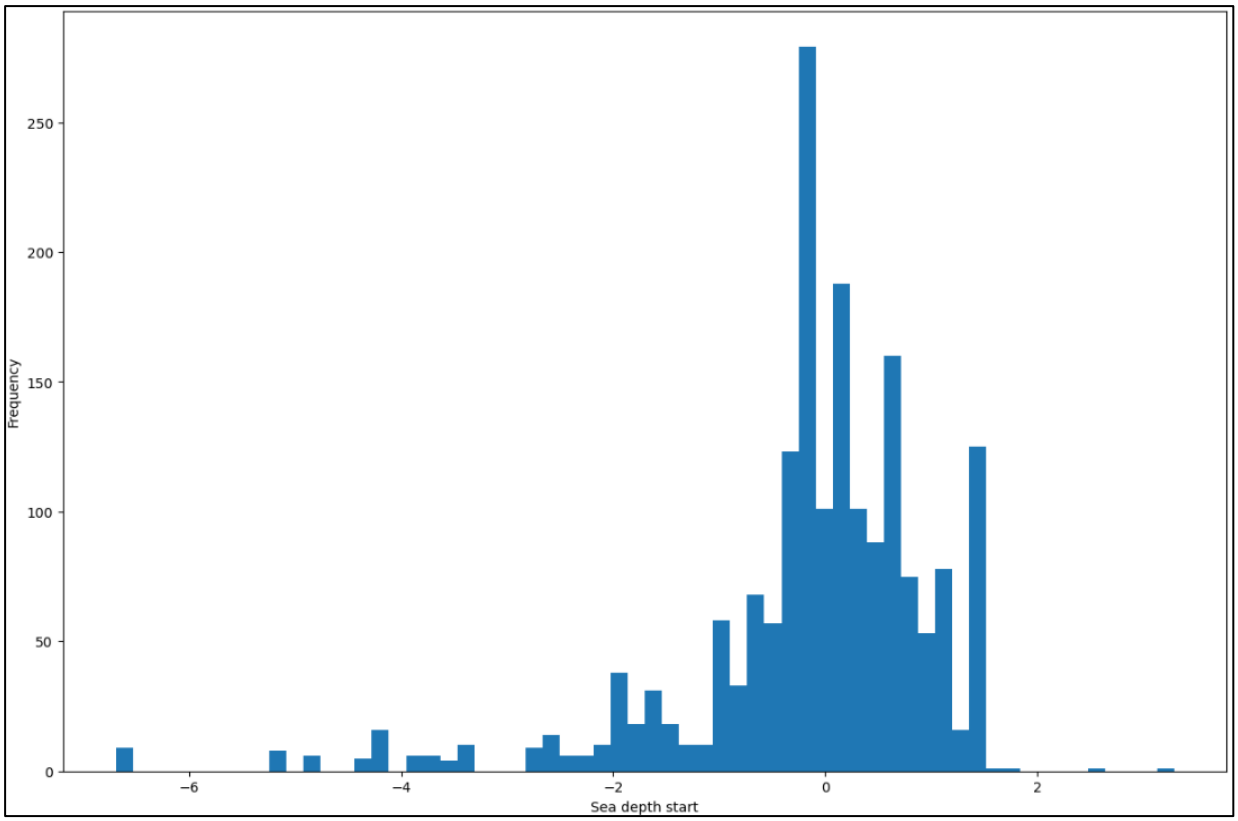


Figure 28: A histogram of sea depth start

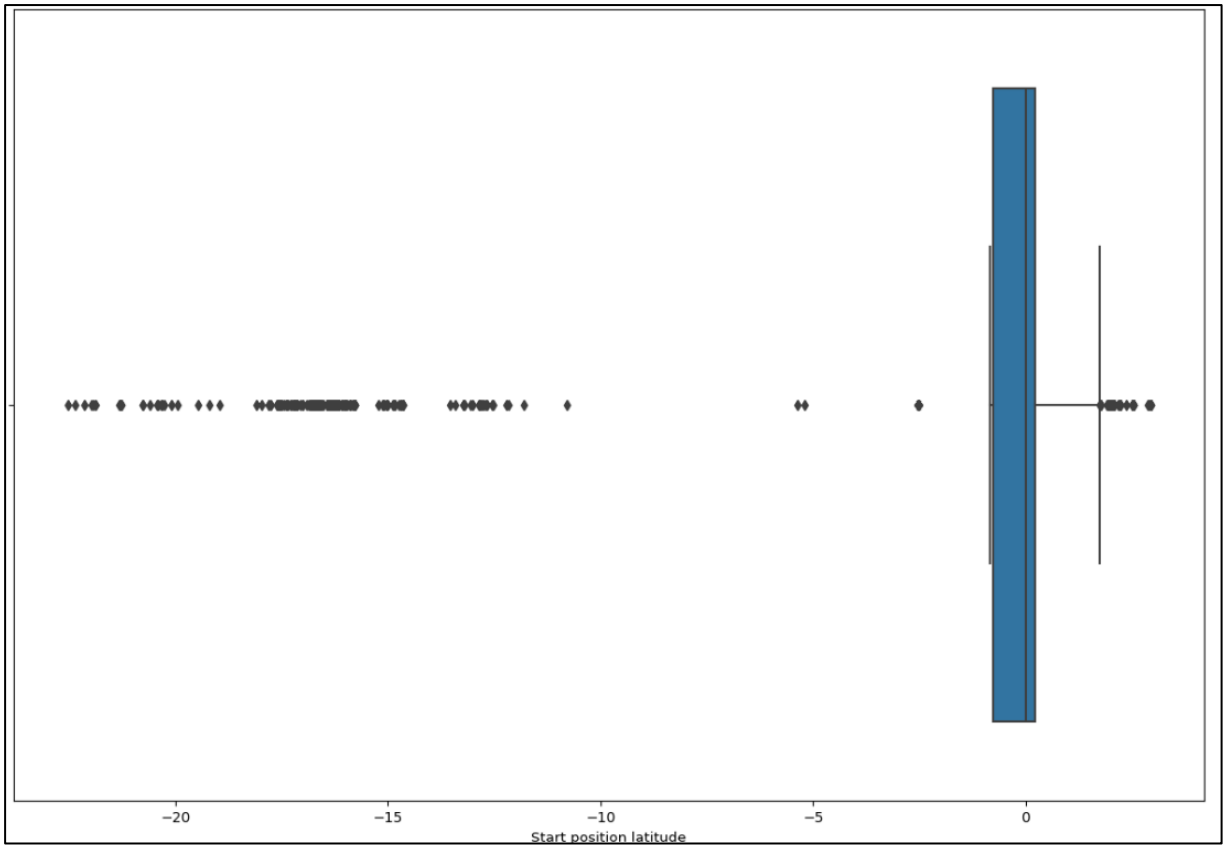


Figure 29: A box plot of start position latitude

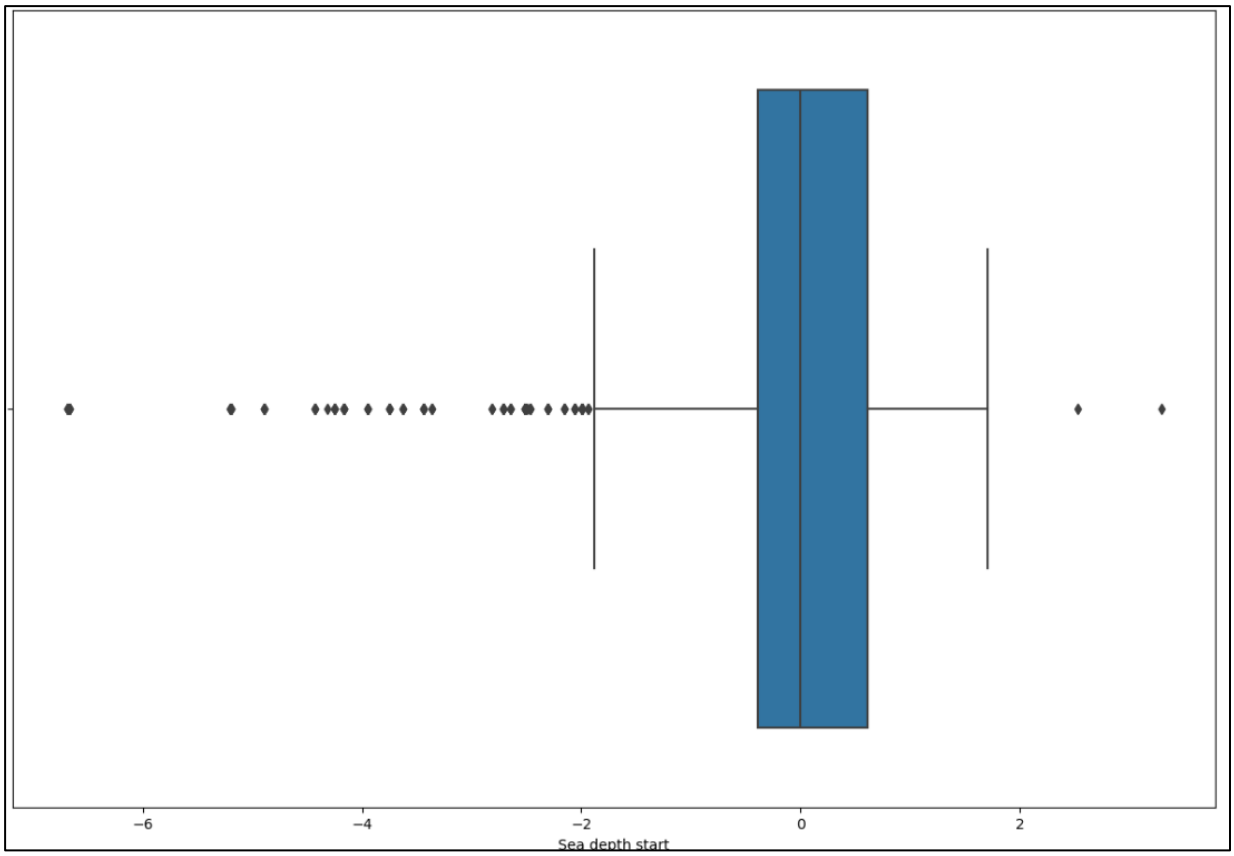


Figure 30: A box plot of sea depth start

3. **Frequency Tables:** Categorical variables are represented on frequency tables to see the overall distribution and frequencies. Potential data issues can be availed through different classes data labels available. By that, we have a better position of understanding categorical variables before hypothesis tests. Figure 31 and 32 respectively; shows a frequency distribution of species available and distribution of any other categorical variable.

```
freq_table
```

Species - main group	col_0 count()
Flatfisk, annen bunnfisk og dypvannsfisk	230
Pelagisk fisk	69
Torsk og torskeartet fisk	1548

Figure 31: Frequency distribution of species main group

```
Frequency table for Message clock time:
col_0                count()
Message clock time
00:02:00                   8
00:03:00                   1
00:22:00                   6
00:25:00                   2
00:45:00                  19
...                      ...
23:53:00                  30
23:54:00                   5
23:55:00                   9
23:56:00                  15
23:57:00                   1

[325 rows x 1 columns]
Frequency table for Radio call signal (ERS):
col_0                        count()
Radio call signal (ERS)
LDQN                             887
LFBC                             960
Frequency table for Vessel name (ERS):
col_0                count()
Vessel name (ERS)
LISE BEATE             887
Trondskjær             960
Frequency table for Activity (code):
col_0              count()
Activity (code)
FIS                   1702
STE                    145
Frequency table for Activity:
col_0      count()
Activity
I fiske       1702
Steaming       145
Frequency table for Start clock time:
col_0                count()
Start clock time
00:00:00                   4
00:02:00                   1
00:06:00                   5
00:07:00                   4
00:10:00                   1
```

Figure 32: Frequency distribution of some categorical variables

The above descriptive statistics by the EDA techniques for numerical and categorical variables can be summarized by the program in figure 32 below:

```
# Compute basic summary statistics for numerical variables
summary_stats = data_df.describe()
print("Summary statistics for numerical variables:")
print(summary_stats)

# Visualize the distribution of numerical variables using histograms
numerical_cols = data_df.select_dtypes(include=['int64', 'float64']).columns
for col in numerical_cols:
    plt.figure(figsize=(15, 10))  # Set figure size to (15, 10)
    plt.hist(data_df[col], bins='auto')
    plt.title(f'Histogram of {col}')
    plt.xlabel(col)
    plt.ylabel('Frequency')
    plt.show()

# Visualize the distribution of numerical variables using box plots
for col in numerical_cols:
    plt.figure(figsize=(15, 10))  # Set figure size to (15, 10)
    sns.boxplot(x=col, data=data_df)
    plt.title(f'Box plot of {col}')
    plt.show()

# Create frequency tables for categorical variables
categorical_cols = data_df.select_dtypes(include=['object']).columns
for col in categorical_cols:
    freq_table = pd.crosstab(index=data_df[col], columns='count()')
    print(f"Frequency table for {col}:")
    print(freq_table)
```

Figure 33: A program for visualizing the distribution of numerical and categorical variables by EDA

4.2.5 Data Visualization

Data Visualization uses univariate, bivariate and multivariate methods to identify more complex relationship in different dimensions of our data. The following subtasks are performed in data visualization stage stating their significant roles:

1. **Univariate Analysis:** Variables are reached out to at point basis by creating kernel density plots, box plots for our numerical variables. For categorical variables pie charts and bar plots are used. Anomalies in data frame can be identified.
2. **Bivariate Analysis:** Two variables are compared to each other to see correlations and dependent relationships between them. For numerical variables scatter plots and line plots for time series samples. For categorical variables, group bar plots are used.
3. **Multivariate Analysis:** The relationship of a number of variables can be analyzed through spearman correlation heatmap and parallel coordinate plots to identify more difficult interactions. Heatmaps produces a correlation between two variables whilst parallel pair plots necessitate a relationship between multiple variables. Cluster correlations can be identified within the dataset.

An overall base map was developed to show our fishing locations from our given dataset based on start position latitudes and longitudes. This helps to have a better starting point of our fishing

hotspots distribution throughout the season. As the latitude and longitude varies, fishing locations also change. Figure 34 below shows a program that was used to produce the base map for our two fishing vessels.

```
# Create a base map centered around a specific location (you can change the latitude and longitude as needed)
base_map = folium.Map(location=[69.5, 17.5], zoom_start=6)

# Add markers for each fishing location based on the start positions
for index, row in data_df.iterrows():
    start_latitude = row['Start position latitude']
    start_longitude = row['Start position longitude']
    species = row['Species FAO']  # You can customize the marker's popup content with other variables

    # Add a marker to the map
    folium.Marker([start_latitude, start_longitude], popup=species).add_to(base_map)

# Save the map as an HTML file
base_map.save('fishing_locations.html')
```

Figure 34: A program to create our starting base map

The distribution over the season along the Norwegian sea along which our two vessels in question operates. The outcome of the process is summarized in figure 35 as follows:

Figure 35: A spatial distribution of fishing hotspots using starting latitude and longitude

4.2.6 Feature Selection and Engineering

Implementation of feature selection and engineering is important, where we are looking at selecting the most relevant variables or features in our given dataset and transform them with the overall goal of improving our model performance, reduction of overfitting by removing unnecessary parameters and thereby increasing interpretability of our model.

4.2.6.1 Variable Analysis

Based on variables provided in our study, here are some potential features we can use to predict location of fish and generate forecasts for our two vessels:

1. **Relevant year**: This variable represents the year in which the fishing activities are recorded and can assist in detecting patterns over a period of time which can be variations in fish abundances.
2. **Message timestamp/Message date/Message time**: Attributes like months, time of day or weeks in which fishing activities or certain seasonal patterns transpired.
3. **Vessel name (ERS)**: Represents vessel identification but can be useful in identifying how each vessel is performing towards a high catch effort. However, it might not be directly relevant for predicting fish location.

4. **Activity (code) and Activity**: Data about the kind of fishing methods used for example pursue seines and longlining. This information can be useful in identifying which fishing activities are more preferred and produce results in certain locations or times.
5. **Start position latitude and Start position longitude**: These two variables identify were fishing activities commenced and acts as spatial feature in identifying fish location.
6. **Stop position latitude and Stop position longitude**: These two variables' marks were the fishing activities ended and acts as spatial feature to aid fish location.
7. **Start sea depth**: Sea depth is an influential parameter because various fish species habitats on different levels in oceanic water bodies and can be a feature to predict fish location.
8. **Duration**: This represents the time span when fishing activities occurred and is useful in determining catch per unit efforts. It is also useful in determining fish availability.
9. **Towing distance**: This variable represents the scope of area covered by certain fishing gears and can be used as a feature to predict fish availability.
10. **Gear FAO**: Knowledge of effectiveness of different fishing gear types in various locations and conditions and can be used as a feature to predict fish availability.
11. **Main species** - FDIR (code), Species FAO (code), and Species FAO: These variables provide information about the target species, which is crucial for predicting good fishing locations for specific species. You can use them as features to predict fish location and availability.
12. **Round weight**: This variable represents the catch weight, which can be used as a response variable or target output in your machine learning models to predict good fishing locations based on catch rates.

Additionally, we might want to consider other external variables such as environmental data (sea surface temperature, weather conditions and salinity) or regulatory factors (e.g., fishing quotas, protected areas) to create a more comprehensive model for predicting fish location and generating forecasts.

The below mentioned variables have justifications why they are being **dropped**:

1. **Message ID**: This variable is likely an identifier for each message or record and may not provide useful information for predicting fish location or availability. Message time, Message date, Message clock time: These variables represent the timestamp of the

message or record. However, there is no sufficient backing for them to directly predict fish location, a replacement with more spatial features is encouraged.

2. **Radio call signal (ERS**): This variable represents the identifier for each vessel's radio call signal. There is no sufficient evidence for this variable to directly impact fish location.

3. **Vessel name (ERS)**: This variable represents the name of the fishing vessel there is no relevant contribution in predicting fish location or availability.

4. **Activity (code) or Activity**: These variables denote types of fishing activities and not have direct impact on determining fish location. Specific variables like gear types can aid in knowing which method performs best in which season.

5. **Stop time, stop date, stop clock time**: These variables represent the time when the fishing activity stopped there is no direct impact on the question at hand.

6. **Duration**: This variable represents the duration of the fishing activity. There is dependence on it for predicting fish location can be replaced by other influential variables.

7. **Location stop (code), Area group stop (code**): These variables represent the area group where the fishing activity stopped.

4.2.6.2 Assumption(s)

In our case, with the data provided when predicting fish location, the most suitable target variable would be either “start position latitude” or “start position longitude”. This is because start position latitude and longitude provide precise spatial information, which is crucial for understanding fish distribution. Hence, they directly influence the outcome of our algorithm training. A new data frame is developed with new specified variables which are start position latitude, start position longitude, sea depth-start, stop position latitude, stop position longitude, round weight, location start (code) and location stop (code).

Figure 36 below shows a snippet of our new data frame developed after extracting the target variable and storing it as a separate array.

	Sea depth start	Stop position latitude	Stop position longitude	Round weight	Location start (code)	Location stop (code)	Target Latitude	Target Longitude
0	-186.292009	68.634683	15.366425	8095.642368	24.607948	24.533688	68.617071	15.389508
1	-145.000000	69.950000	16.883000	840.000000	29.000000	29.000000	69.950000	16.950000
2	-145.000000	69.950000	16.883000	75.000000	29.000000	29.000000	69.950000	16.950000
3	-145.000000	69.950000	16.883000	4.000000	29.000000	29.000000	69.950000	16.950000
4	-145.000000	69.950000	16.883000	3.000000	29.000000	29.000000	69.950000	16.950000
...	...	...	...	...	...	...	...	...
1842	0.000000	69.944000	21.628000	54000.000000	24.607948	24.533688	69.941000	21.626000
1843	0.000000	69.964000	21.604000	64000.000000	24.607948	24.533688	69.964000	21.604000
1844	0.000000	69.964000	21.604000	15.000000	24.607948	24.533688	69.964000	21.604000
1845	0.000000	69.964000	21.604000	5.000000	24.607948	24.533688	69.964000	21.604000
1846	-186.292009	68.634683	15.366425	8095.642368	24.607948	24.533688	68.617071	15.389508

1847 rows × 8 columns

Figure 36: A new data frame with set target variables

The program used to select specified variables and setting our target variables is summarized in figure 37 below.

```
# Select the specified variables
selected_columns = [
    'Start position latitude',
    'Start position longitude',
    'Sea depth start',
    'Stop position latitude',
    'Stop position longitude',
    'Round weight',
    'Location start (code)',
    'Location stop (code)'
]

# Create a new DataFrame with the selected columns
new_df = data_df[selected_columns].copy()

# Assuming 'Start position latitude' and 'Start position longitude' are the target variables
# Move them to the last columns
new_df['Target Latitude'] = new_df.pop('Start position latitude')
new_df['Target Longitude'] = new_df.pop('Start position longitude')

# Display the new DataFrame
new_df.head()
```

Figure 37: A program for setting target variables

4.2.7 Correlation Analysis

This stage or experimental plan is critical in addressing objective 2 of our research project. Descriptive statistics analysis was to find how the dataset is distributed before the training phase. The need for a correlation heatmap is to find the strengths of relationship that exist between numerical variables and the target variable involved. Linear and non-linear relationships can also be drawn from a correlation heatmap.

From the heatmap it can be interpreted that that a positive strong correlation exists between numerical variables: stop position latitude and stop position longitude of 0.96, stop position latitude and target longitude of 0.96, stop position longitude and target latitude of 0.95, stop position longitude and target longitude of 0.99. This means that there a dependable relationship between these variables in locating fish locations. When one variable increases, another variable or increases. There is strong negative correlation between numerical variables: sea depth-start and stop position latitude of -0.24, location start (code) of -0.31, location stop (code) of -0.33, target latitude and sea depth start of -0.25. This means the relationship cannot be relied on in trying to know features that directly impact fish location. When one variable increases, the other variable is decreasing. An overall view notes the existence of a positive correlation between independent and dependent variables for this dataset. It is also advisable to find to which degree our predictor variables have a multicollinearity relationship in order to increase

the accuracy of our predictive models through calculating the variance factor. Figure 38 below shows the correlation matrix produced by variables in our dataset.

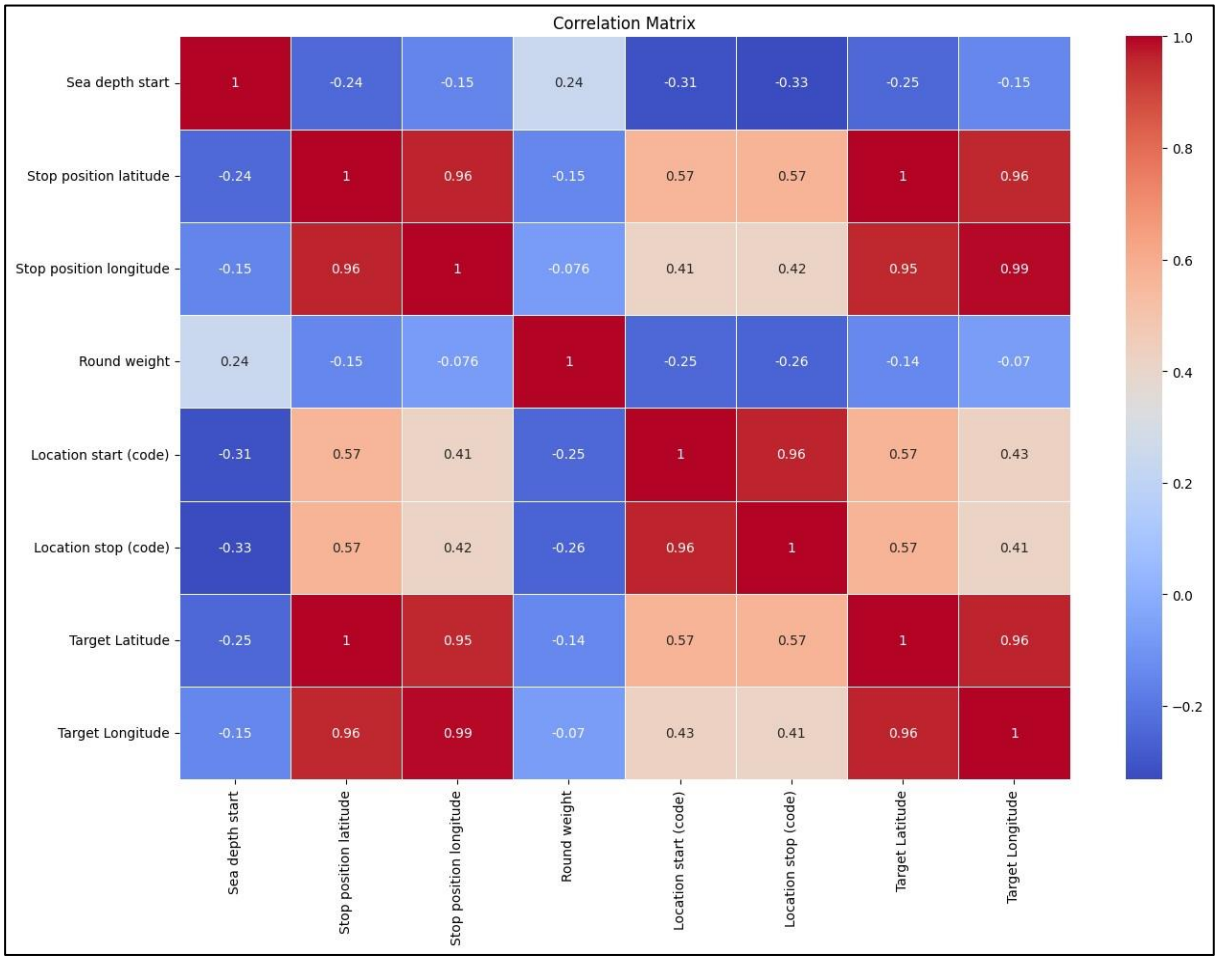


Figure 38:Spearman correlation heatmap with correlation coefficients of numerical variables from our dataset

Scatter plots and line plots (assuming our data is time series) can also be created to identify possible form of relationship between our numerical variables showing individual data points. For the scatter plot and line plot, each variable was plotted against target latitude and target longitude. The data showed a good fit to the available algorithms since small residual values can be noted along, the regression line. Figure 44 and figure 45 below summarizes relationships produced by scatter plot and line plot.

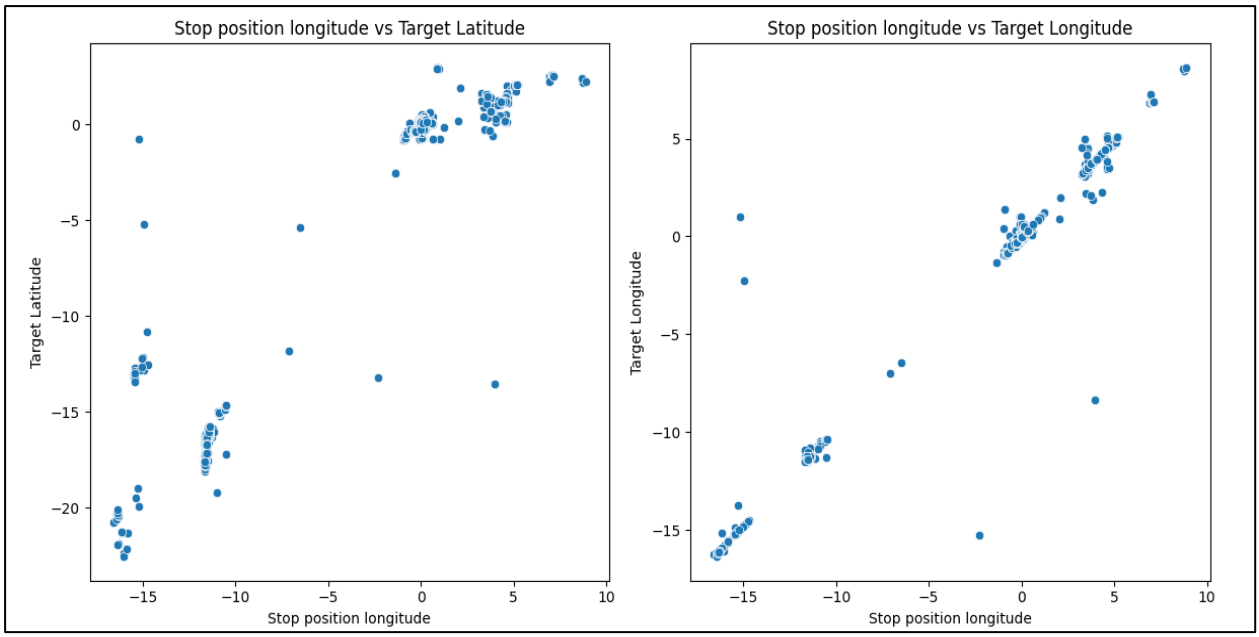


Figure 39: A scatter plot for target variables vs stop position longitude

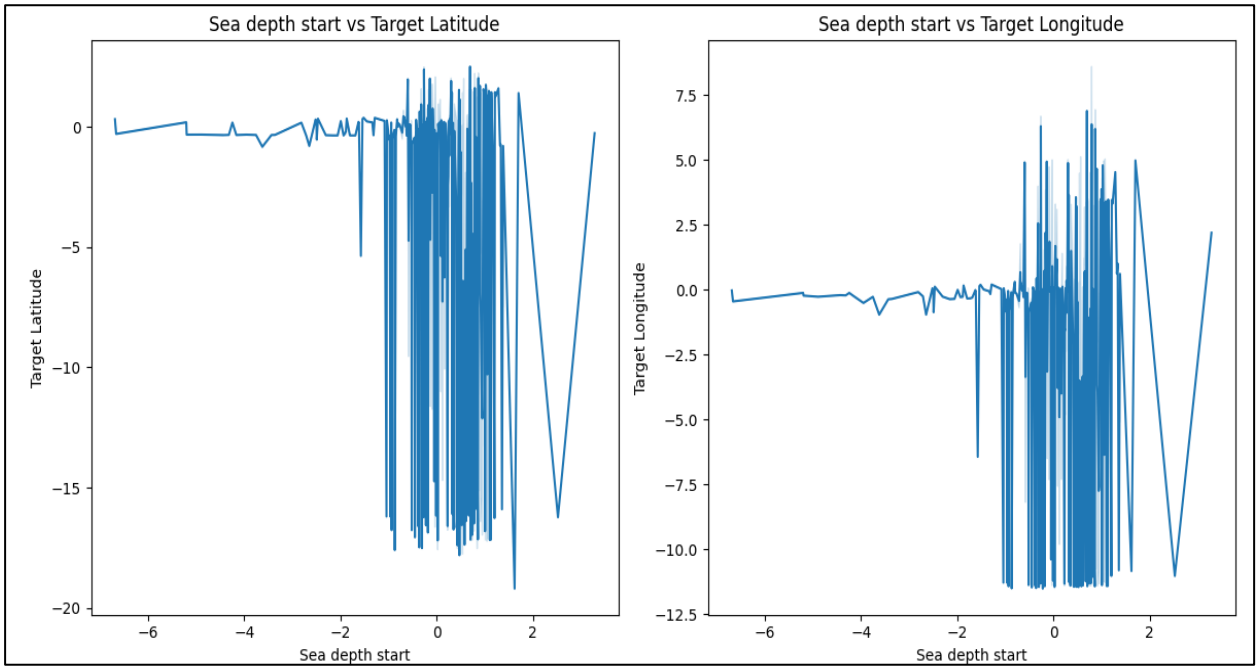


Figure 40: A line plot for target variables vs sea depth start

4.2.8 Train regression model

This stage involves training our available dataset with our regression algorithms. This stage is key to in addressing the train and test of multioutput regression algorithms, in Objective 1, in predicting fish location. Data frame is split into two sets which are training set and testing set (20% of our dataset). The training set trains the model whilst the testing set is used for performance evaluation of our chosen models. A random state 42 was chosen to ensure the results are continuously produced on repeated processes like run, shuffling, splitting and evaluations when running different algorithms. The regression models are denoted Model 1 Decision Trees, Model 2 Support Vector Machines, Model 3 Random Forest, Model 4 Gradient Boosting Regression and Model 5 Linear Regression.

For models, Gradient Boosting Regression and Random Forest the parameter n-estimators 100 is used which is an arbitrary number to represent our base models in ensemble learning techniques like in our case RF and GBR.

The following programs in figures 41, 42, 43, 44 and 45 below were used to perform train, test split and to produce predictions on our features in respect to target variables, target latitude and target longitude.

```
# Model 1: MultiOutputRegressor with Decision Trees (DT)
model_dt = MultiOutputRegressor(DecisionTreeRegressor(random_state=42))
model_dt.fit(X_train, y_train)
rmse_dt_lat, rmse_dt_lon, mae_dt_lat, mae_dt_lon, r2_dt_lat, r2_dt_lon = evaluate_model(model_dt, X_test, y_test)
print("\nModel 1 (Decision Trees) Performance:")
print("Target Latitude:")
print("RMSE:", rmse_dt_lat)
print("MAE:", mae_dt_lat)
print("R^2:", r2_dt_lat)
print("Target Longitude:")
print("RMSE:", rmse_dt_lon)
print("MAE:", mae_dt_lon)
print("R^2:", r2_dt_lon)
```

Figure 41: A snippet train and split program for Model 1

```
# Model 2: MultiOutputRegressor with Support Vector Machines (SVM)
model_svm = MultiOutputRegressor(SVR())
model_svm.fit(X_train, y_train)
rmse_svm_lat, rmse_svm_lon, mae_svm_lat, mae_svm_lon, r2_svm_lat, r2_svm_lon = evaluate_model(model_svm, X_test, y_test)
print("\nModel 2 (Support Vector Machines) Performance:")
print("Target Latitude:")
print("RMSE:", rmse_svm_lat)
print("MAE:", mae_svm_lat)
print("R^2:", r2_svm_lat)
print("Target Longitude:")
print("RMSE:", rmse_svm_lon)
print("MAE:", mae_svm_lon)
print("R^2:", r2_svm_lon)
```

Figure 42:A snippet program for train and split for Model 2

```
# Model 3: MultiOutputRegressor with Random Forest (RF)
model_rf = MultiOutputRegressor(RandomForestRegressor(n_estimators=100, random_state=42))
model_rf.fit(X_train, y_train)
rmse_rf_lat, rmse_rf_lon, mae_rf_lat, mae_rf_lon, r2_rf_lat, r2_rf_lon = evaluate_model(model_rf, X_test, y_test)
print("\nModel 3 (Random Forest) Performance:")
print("Target Latitude:")
print("RMSE:", rmse_rf_lat)
print("MAE:", mae_rf_lat)
print("R^2:", r2_rf_lat)
print("Target Longitude:")
print("RMSE:", rmse_rf_lon)
print("MAE:", mae_rf_lon)
print("R^2:", r2_rf_lon)
```

Figure 43: A snippet program for train and split for Model 3

```
# Model 4: MultiOutputRegressor with Gradient Boosting Regression (GBR)
model_gbr = MultiOutputRegressor(GradientBoostingRegressor(n_estimators=100, random_state=42))
model_gbr.fit(X_train, y_train)
rmse_gbr_lat, rmse_gbr_lon, mae_gbr_lat, mae_gbr_lon, r2_gbr_lat, r2_gbr_lon = evaluate_model(model_gbr, X_test, y_test)
print("\nModel 4 (Gradient Boosting Regression) Performance:")
print("Target Latitude:")
print("RMSE:", rmse_gbr_lat)
print("MAE:", mae_gbr_lat)
print("R^2:", r2_gbr_lat)
print("Target Longitude:")
print("RMSE:", rmse_gbr_lon)
print("MAE:", mae_gbr_lon)
print("R^2:", r2_gbr_lon)
```

Figure 44: A snippet program for train and split for Model 3

```
# Model 5: MultiOutputRegressor with Linear Regression (LR)
# Note: Logistic Regression is not suitable for regression tasks. Using Linear Regression instead.
model_lr = MultiOutputRegressor(LinearRegression())
model_lr.fit(X_train, y_train)
rmse_lr_lat, rmse_lr_lon, mae_lr_lat, mae_lr_lon, r2_lr_lat, r2_lr_lon = evaluate_model(model_lr, X_test, y_test)
print("\nModel 5 (Linear Regression) Performance:")
print("Target Latitude:")
print("RMSE:", rmse_lr_lat)
print("MAE:", mae_lr_lat)
print("R^2:", r2_lr_lat)
print("Target Longitude:")
print("RMSE:", rmse_lr_lon)
print("MAE:", mae_lr_lon)
print("R^2:", r2_lr_lon)
```

Figure 45:A snippet program for train and split for Model 5

4.2.9 Evaluation Metrics

To compare and contrast individual performances of five regression algorithms and selecting the best algorithm to meet objective 3 and 4 this stage was implemented. Typical regression problems evaluation metrics are Mean Absolute Error (MAE), Root Mean Square Error (RMSE) and coefficient of determination (R-squared) etc. These were used to evaluate the performance of our models, a broader range for comparison between models to see which offers better predictability of our set target variables. Below is a set of equations that represent each evaluation metric.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - y_2|$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i^- y_2)^2}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - y_2)^2}{\sum_{i=1}^{n}(y_i - y_o)^2}$$

The equation above represents a series of observations *N*, with predicted value of y_i of the *ith* observation and y_2 corresponds to the actual value. In context of our sampled actual data $y_{o.}$ denotes the mean. MAE is that absolute difference between the actual observations and the predicted output. It generalizes the magnitude of error in our series of predictions. It can also be termed a smoothing process of errors in training outliers hence its outcome is a generic performance measurement for a given model. RMSE is different in operation with MAE, because it penalizes the existence of large errors or outliers in case predicted values deviates much from the set target values. R-squared, known as the coefficient of determination, which can be a value from negative infinity to 1 represents the fit of our prediction model with the ground truth. Various possibilities are for $R^2 \geq 0$, considering a linear regression with no constraints COD is negative. For R^2 = 0, If two independent variables exist on this fitted line or horizontal line, they are considered uncorrelated. It is noted that behaviors of COD show an independent nature with the linearity of a regression model meaning to say R^2 can be very high for a non-linear model and R^2 can be very low for a linear regression model [88].

Figure 46 below shows a summary of results in our regression analysis evaluation using MAE, RMSE and R^2 metrics for target latitude and target longitude.

	RMSE Latitude	RMSE Longitude	MAE Latitude	MAE Longitude	R^2 Latitude	R^2 Longitude
Decision Trees (DT)	0.423630	1.081322	0.060584	0.106972	0.994502	0.942877
Support Vector Machines (SVM)	0.855552	0.872279	0.209103	0.214038	0.977575	0.962828
Random Forest (RF)	0.652303	1.026052	0.081178	0.111339	0.986964	0.948567
Gradient Boosting Regression (GBR)	0.621512	0.998098	0.107505	0.137989	0.988166	0.951331
Linear Regression (LR)	0.567945	0.831824	0.124241	0.170607	0.990118	0.966196

Figure 46: Detailed evaluation metrics of RSME, MAE and CoD

The program that was used to write the metrics keys is summarized in figure 47 below.

```
# Define the metrics dictionary and columns list
metrics = {
    'Decision Trees (DT)': [rmse_dt_lat, rmse_dt_lon, mae_dt_lat, mae_dt_lon, r2_dt_lat, r2_dt_lon],
    'Support Vector Machines (SVM)': [rmse_svm_lat, rmse_svm_lon, mae_svm_lat, mae_svm_lon, r2_svm_lat, r2_svm_lon],
    'Random Forest (RF)': [rmse_rf_lat, rmse_rf_lon, mae_rf_lat, mae_rf_lon, r2_rf_lat, r2_rf_lon],
    'Gradient Boosting Regression (GBR)': [rmse_gbr_lat, rmse_gbr_lon, mae_gbr_lat, mae_gbr_lon, r2_gbr_lat, r2_gbr_lon],
    'Linear Regression (LR)': [rmse_lr_lat, rmse_lr_lon, mae_lr_lat, mae_lr_lon, r2_lr_lat, r2_lr_lon]
}

columns = ['RMSE Latitude', 'RMSE Longitude', 'MAE Latitude', 'MAE Longitude', 'R^2 Latitude', 'R^2 Longitude']

# Create DataFrame from metrics dictionary
metrics_df = pd.DataFrame(metrics.values(), index=metrics.keys(), columns=columns)

# Display the DataFrame
metrics_df
```

Figure 47: A snippet program for calling evaluation metrics

4.2.10 Discussion of Results

This stage seeks to discuss the outcomes of our evaluation metrics for the purpose of choosing a highly performing regression machine learning model for future predictions of fish locations. Based on the provided performance metrics, here is an interpretation of the results of each model:

4.2.10.1 Model 1 (Decision Trees)

1. Target Latitude:
    - RMSE (Root Mean Square Error): 0.424
        - The model has relatively low RMSE, indicating that the residuals or prediction errors have a smaller magnitude.
    - MAE (Mean Absolute Error): 0.061
        - The model has a low MAE which means that the average magnitude of errors is small.
    - R^2 (Coefficient of Determination): 0.995
        - The model has a high R^2 score, indicating that a significant portion of the variance in the target latitude can be explained by the independent or numerical variables.
2. Target Longitude
    - RSME: 0.989
        - The model has a relatively low RMSE indicating that the residuals or predicted errors have a smaller magnitude.
    - MAE: 0.107
        - The model has a low MAE, which means that the average magnitude of errors is small.
    - R^2 (Coefficient of Determination): 0.943

- The model has a high R^2 score, indicating that a significant proportion of the variance in the target longitude can be explained by the independent or numerical variables.

4.2.10.2 Model 2 (Support Vector Machines)

1. Target Latitude:
    - RMSE: 0.856
        - The model has a higher RMSE compared to Model 1, indicating that the residuals or predicted errors have a large magnitude.
    - MAE: 0.209
        - The model has a higher MAE compared to Model 1, which means that the average magnitude of errors is larger.
    - R^2: 0.978
        - The model has a lower R^2 score compared to Model 1, indicating a smaller portion of the variance in the target latitude can be explained by the independent variables.
2. Target Longitude
    - RMSE: 0.872
        - The model has a higher RMSE compared to Model 1, indicating that the residuals or prediction errors have a large magnitude.
    - MAE: 0.214
        - The model has a higher MAE compared to Model 1, which means that the average magnitude of the errors is larger.
    - R^2: 0.963
        - The model has a higher R^2 score compared to Model 1, indicating that a larger portion of the variance in the target longitude can be explained by the independent variables.

4.2.10.3 Model 3 (Random Forest)

1. Target Latitude:
    - RMSE: 0.652
        - The model has a lower RSME compared to Model 2, indicating that the residuals or prediction errors have a smaller magnitude.
    - MAE: 0.081
        - The model has a lower MAE compared to Model 2, which means that the average magnitude of the errors is smaller.
    - R^2: 0.987
        - The model has a higher R^2 score as compared to Model 2, indicating the larger proportion of the variance in the target latitude can be explained by the independent variables.
2. Target Longitude:
    - RMSE: 1.026
        - The model has a higher RMSE compared to Model 1 and Model 2, indicating that the residuals or prediction errors have a larger magnitude.
    - MAE: 0.111

- The model has a higher MAE compared to Model 1 which means that the average magnitude of errors is larger but not as compared to Model 2.
- R^2: 0.948
  - The model has a lower R^2 score compared to Model 1 and Model 2, indicating that a smaller portion of variance in the target longitude can be explained by the independent variables.

4.2.10.4 Model 4 (Gradient Boosting Regression)

1. Target Latitude:
   - RMSE: 0.622
     - The model has lower RMSE as compared to Model 2 and Model 3, indicating that the residuals or prediction errors have a smaller magnitude.
   - MAE: 0.108
     - The model has a lower MAE than Model 2 and slightly higher than Model 3 which means that the average magnitude of the errors is smaller.
   - R^2: 0.988
     - The model has a higher R^2 score compared to Model 2 and Model 3, indicating that a larger portion of the variance in the target latitude can be explained by independent variables.
2. Target Longitude:
   - RMSE: 0.996
     - The model has a lower RMSE compared to Model 2 and Model 3, indicating that the residuals or prediction errors have a smaller magnitude.
   - MAE: 0.138
     - The model has a higher MAE compared to Model 1, which means that the average magnitude of the errors is larger. However, it has a lower MAE compared to Model 2 and Model 3.
   - R^2: 0.951
     - The model has higher R^2 score as compared to Model 1 and Model 3, indicating that the larger portion of the variance in the target longitude can be explained by the independent variables.

4.2.10.5 Model 5 (Linear Regression)

1. Target Latitude:
   - RMSE: 0.568
     - The model has a lower RSME compared to Model 2, Model 3 and Model 4, indicating that the residuals or prediction errors have a smaller magnitude.
   - MAE: 0.124
     - The model has a lower MAE compared to Model 2 which means that the average magnitude of the errors is smaller. However, the model has higher MAE value than Model 1, 3 and 4.
   - R^2: 0.990

    - The model has a higher than Model 2, Model 3 and Model 4, indicating that the larger portion of the variance in the target latitude can be explained by the independent variables.

2. Target Longitude:
    - RMSE: 0.832
        - The model has a lower RSME compared to Model 2, Model 3, Model 4, indicating that the residuals or prediction errors have a smaller magnitude.
    - MAE: 0.171
        - The model has higher MAE compared to Model 1, Model 3 and Model 4, which means that average magnitude of the errors is larger.
    - R^2 : 0.966
        - The model has a higher R^2 score compared to Model 2, Model 3 and Model 4, indicating that a larger portion of the variance in the target longitude can be explained by the independent variables.

In summary, based on the provided performance metrics, model 1 (Decision Trees) and Model 5 (Linear Regression) have the best overall performance for predicting both target latitude and target longitude. Model 5 (Linear Regression) has slightly better performance metrics compared to Model 1 (Decision Trees) for predicting target longitude. However, Model 1(Decision Trees) has slightly better performance metrics for predicting target latitude. The choice between these two models depends on the specific requirements of the application needed and trade-offs that a firm is willing to make between model complexity and performance.

Based on the results provided, Model 3 (Random Forest) and Linear Regression (Model 5) have the best overall performance for predicting both target latitude and target longitude. But since we intend to adopt a model that analyzes complex relations and interactions between features, that includes future training of model with new dataset or numerical variables (environmental data like salinity, sea bottom temperature dissolved oxygen content, nitrate-nitrogen content, migration and spawning patterns etc.) to enhance predictability of fish locations; a Random Forest machine learning model performs better. It can be saved and loaded on pickle.

To develop a Random Forest model, hyperparameter tuning through grid search is performed in a bid to ascertain the best hyperparameters for our model. Figure 46 below shows a program developed to have an optimized random forest model.

```
# Define the Random Forest model
rf_reg = RandomForestRegressor(random_state=42)

# Set up the hyperparameter grid for tuning
param_grid = {
    'n_estimators': [100, 200, 300],
    'max_depth': [10, 20, 30, None],
    'min_samples_split': [2, 5, 10],
    'min_samples_leaf': [1, 2, 4]
}

# Perform grid search for hyperparameter tuning
grid_search = GridSearchCV(rf_reg, param_grid, cv=5, scoring='neg_mean_squared_error', n_jobs=-1)
grid_search.fit(X_train, y_train)

# Get the best hyperparameters
best_params = grid_search.best_params_
print("Best hyperparameters:", best_params)

# Create the optimized Random Forest model with the best hyperparameters
optimized_rf_reg = RandomForestRegressor(**best_params, random_state=42)

# Train the optimized Random Forest model
optimized_rf_reg.fit(X_train, y_train)

# Save the optimized Random Forest model
with open('optimized_rf_reg.pkl', 'wb') as f:
    pickle.dump(optimized_rf_reg, f)

# Load the saved model
with open('optimized_rf_reg.pkl', 'rb') as f:
    loaded_optimized_rf_reg = pickle.load(f)
```

```
Best hyperparameters: {'max_depth': 10, 'min_samples_leaf': 2, 'min_samples_split': 2, 'n_estimators': 200}
```

Figure 48: A snippet program for hyperparameter tuning for Random Forest Model

After that, it is important that we develop a code that will make new predictions or forecasting on new data. A saved model is loaded from a file using pickle serialization. Then new data in form of csv files is loaded, preprocessing occurs if it is necessary depending on nature of data using the same procedure of identifying and handling outliers, missing values, removing duplicates, feature scaling and identifying data types (categorical or numerical). This ensures more reliability and integrity of our data.

It is important to preprocess the new data in the same way as the training data to ensure that the model can make accurate predictions. To load a saved model from a file and make new predictions on new data we can use the following code in figure below. The similar code can be used for other saved models or new datasets to make predictions to allow us to apply trained models to practical situations and then evaluate their performance on totally new data.

```
# Load the saved model from a file
with open('optimized_rf_reg.pkl', 'rb') as f:
    optimized_rf_reg = pickle.load(f)

# Load the new data
new_data = pd.read_csv('new_data.csv')

# Preprocess the new data (if necessary)
new_data = preprocess_data(new_data)

# Make predictions on the new data using the optimized model
new_predictions = optimized_rf_reg.predict(new_data)

# Print the predictions
print('New Predictions:', new_predictions)
```

Figure 49: A program code for future predictions using optimized model

5 Conclusion and Future Work

5.1 Conclusion

This project is a major breakthrough in the fisheries industry. It is a clear demonstration on how companies can come up with important sustainable, data-centric decisions and strategies on selecting the best fishing methods, knowing fish spawning patterns, fishing zones, reduction of environmental damage by preserving non-target species, and generally an optimized/cost effective approach.

The author of this research performed a comparison over different machine learning regression algorithms (chosen concept 1) through regression evaluation metrics Mean Absolute Error (MAE), Coefficient of determination (R^2) and Root Mean Square Error (RMSE) to determine the accuracy of predicting fish location. High performance was on noted on two regressor algorithms, Linear Regression and Random Forest but since the focus was on a model that can handle sparse data variables and handles large datasets, Random Forest model was preferred and is regarded high performing. The overall accuracy of prediction is greatly influenced by handling non-linearities in available dataset, enhanced feature selection so as to indicate how much each feature can contribute to the model's performance.

Over the years, a lot of companies have been gathering satellite derived big marine data variables in form of oceanographic data, environmental data and AIS data but with not much knowledge on how to utilize it in drawing useful correlational relationships towards a higher catch effort. In general, numerical variables played a significant role as predictor variables in predicting fish location hotspots represented target latitude vs target longitude in correlation analysis. To mention a few, location start and stop codes, start and stop latitudes and longitudes, round weight representing catch data as reflected by Pearson correlation heatmap.

5.2 Future Work

The idea of future work is of great importance since it directs virgin direction of breakthrough in research. Our main focus was predicting the location where a catch of species is high, but however through using machine learning models in fishing industries the following measures and contributions when handling similar projects can be suggested:

1. **Datasets:** In as much as high performance was noted on our models, further improvement is needed. Adding environmental predictor variables (assumed to be of great impact towards fish availability) like sea surface temperatures, oceanic currents, sea salinity, chemical oxygen demand, chlorophyll, sea surface heights, eddy current energy amongst others offers a greater ability in predicting fish abundance and location. It is important to have variables form a specific spatial area or scale to avoid overlooking patterns that is available in variables by assuming average data for a large sea area spanning kilometers. The need to explore data form different fishing methods for example pursue seines and trawling.
2. **Deep learning techniques for prediction:** It is important to appreciate the use of deep learning algorithms like autoencoders, Long Short-Term Memory (LSTM). For example, the use LSTM networks works greatly when handling sequential data in form of oceanographic and environmental variables, migration patterns. It overcomes with easiness the issue gradient vanishing and avoidance of long dependency issues. In most cases, sparse data availability elevates the model accuracy.
3. **Training Issues:** Despite good overall performance of regressor algorithms, we used a small dataset only for the 2023 season. This was due to some data availability issues and execution time frame. It is advisable to train-test using data for several years taking advantage of the longitudinal test-train regime to improve model accuracy. The question at hand can also be handled as a classification problem, it will of keen interest to compare how regressors and classifiers handle this problem using metrics ROC analysis, AUC, confusion matrix, precision and recall.
4. **Integration with IOT systems:** Real time monitoring of sea parameters from sensory hardware mechanism can be integrated with machine learning to amass data automatically so that prediction forecasting for future conditions or seasons can be done from sensory data.

Works cited

[1] H. Tercan and T. Meisen, "Machine learning and deep learning based predictive quality in manufacturing: a systematic review," Journal of Intelligent Manufacturing, vol. 33, no. 7, pp. 1879-1905, 2022/10/01 2022, doi: 10.1007/s10845-022-01963-8.

[2] H. Tercan, A. Guajardo, and T. Meisen, "Industrial Transfer Learning: Boosting Machine Learning in Production," in 2019 IEEE 17th International Conference on Industrial Informatics (INDIN), 22-25 July 2019 2019, vol. 1, pp. 274-279, doi: 10.1109/INDIN41052.2019.8972099.

[3] M. V. N. L. Chaitanya, S. Chinni, and V. Jyothi, "The Importance Of Quality Management System For a Successful Health Care Industry": *A Review Based on Case Studies*, 2018.

[4] S. Schorr, M. Möller, J. Heib, and D. Bähre, "Quality Prediction of Drilled and Reamed Bores Based on Torque Measurements and the Machine Learning Method of Random Forest," *Procedia Manufacturing*, vol. 48, pp. 894-901, 2020/01/01/ 2020, doi: https://doi.org/10.1016/j.promfg.2020.05.127.

[5] M. Arikkök, Total Quality Management, 2017.

[6] F. Marimon, N. Melao, and R. Bastida, "Motivations and benefits of quality management systems in social services: mediation of the implementation process," *Total Quality Management and Business Excellence*, vol. 32, 06/07 2019, doi: 10.1080/14783363.2019.1626707.

[7] D. Ross, "The challenges of implementing a quality management system in contemporary large scale construction projects," 2021.

[8] T. Gittler et al., "Towards predictive quality management in assembly systems with low quality low quantity data – a methodological approach," Procedia CIRP, vol. 79, pp. 125-130, 2019/01/01/ 2019, doi: https://doi.org/10.1016/j.procir.2019.02.026.

[9] D. C. Montgomery, Wiley, "Introduction to Statistical Quality Control, 2020.

[10] J. Frick and P. Grudowski, "Quality 5.0: A Paradigm Shift Towards Proactive Quality Control in Industry 5.0," *International Journal of Business Administration*, vol. 14, p. 51, 06/19 2023, doi: 10.5430/ijba.v14n2p51.

[11] Killarney, "European Inland Fisheries and Aquaculture Advisory Commission," presented at the EIFAAC Symposium on Inland Fisheries and Aquaculture: Advances in Technology, *Stock Assessement and Citizen Science in an era of Climate Change, Ireland*, 2022.

[12] M. Krishnan, "Data analytics in fisheries," 2023, pp. 135-150.

[13] K. M. Stehfest, T. A. Patterson, L. Dagorn, K. N. Holland, D. Itano, and J. M. Semmens, "Network analysis of acoustic tracking data reveals the structure and stability of fish aggregations in the ocean," *Animal Behaviour*, vol. 85, no. 4, pp. 839-848, 2013/04/01/ 2013, doi: https://doi.org/10.1016/j.anbehav.2013.02.003.

[14] A. Sohns, G. M. Hickey, and O. Temby, "Exploring the potential impacts of machine learning on trust in fishery management," *Fish and Fisheries*, vol. 23, no. 4, pp. 1016-1023, 2022/07/01 2022, doi: https://doi.org/10.1111/faf.12658.

[15] V. Klemas, "Coastal and Environmental Remote Sensing from Unmanned Aerial Vehicles: An Overview," *Journal of Coastal Research*, vol. 315, pp. 1260-1267, 09/02 2015, doi: 10.2112/JCOASTRES-D-15-00005.1.

[16] X. Lin, N. Sanket, N. Karapetyan, and Y. Aloimonos, OysterNet, "Enhanced Oyster Detection Using Simulation", 2022.

[17] M. Bai J and M. Duke, "Statistical process control," International Journal of Commerce and Management, vol. 06, 02/04 2019.

[18] J. S. Oakland, Statistical Process Control, 6th ed. 2 Park Square, Milton Park, Abingdon, Oxon OX14 4RN: Routledge, 2008.
[19] J. M. Juran, McGraw Hill and A. B. Godfrey, "Juran's Quality Handbook", 1999.
[20] T. K. Ross, "A statistical process control case study," (in eng), *Quality Management Health Care*, vol. 15, no. 4, pp. 221-36, Oct-Dec 2006, doi: 10.1097/00019514-200610000-00004.
[21] X. a. Goh, "Statistical Techniques for quality," The TQM Magazine, vol. 11 no 4, pp. 238-41, 1999.
[22] M. Xie and T. N. Goh, "Statistical techniques for quality," *The TQM Magazine*, vol. 11, no. 4, pp. 238-242, 2004.
[23] A.-R. S. e. al, "Implementation of Statistical Process Control Techniques to reduce the Defective Ratio," *Proceedings of the International Conference of Industrial Engineering and Operations Management* p. 10, 2018.
[24] T. Dasgupta, "Maximizing the effectiveness of control charts: A framework for reacting to out-of-control signals," *Proceedings of Annual Quality Congress*, vol. 57 No 0, pp. 327-337, 2003.
[25] P. A.C, "Discussion on "Controversies and contradictions in Statistical Process Control," *Journal of Quality Technology*, vol. 32 No 4, no. ISSN 0022-4065, 2000.
[26] D. Noskievicova, "Effective Implementation of Statistical Process Control " *Engineering the Future*, 2010.
[27] P. Gejdos, "Continuous Quality Improvement by Statisitcal Process Control " *Business Economics and Managment 2015 Conference*, BEM2015, pp. 564-5-572, 2015.
[28] M. V. Koutras, S. Bersimis, and P. E. Maravelakis, "Statistical Process Control using Shewhart Control Charts with Supplementary Runs Rules," *Methodology and Computing in Applied Probability*, vol. 9, no. 2, pp. 207-224, 2007/06/01 2007, doi: 10.1007/s11009-007-9016-8.
[29] M. Terek and Ľ. Hrnčiarová, Štatistické riadenie kvality. IURA EDITION, spol. s ro, 2004.
[30] M. Noman et al., "A Model for Maintenance Planning and Process Quality Control Optimization Based on EWMA and CUSUM Control Charts," *Transactions of famena*, vol. 45, 04/28 2021, doi: 10.21278/TOF.451021920.
[31] M. Aslam, M. Azam, and C.-H. Jun, "A new control chart for exponential distributed life using EWMA," *Transactions of the Institute of Measurement and Control*, vol. 37, no. 2, pp. 205-210, 2015/02/01 2014, doi: 10.1177/0142331214537293.
[32] I. Gomes, S. Mingoti, and C. Oliveira, "A novel experience in the use of control charts for the detection of nosocomial infection outbreaks," Clinics (São Paulo, Brazil), vol. 66, pp. 1681-9, 09/30 2011, doi: 10.1590/S1807-59322011001000004.
[33] A. R. Motorcu and A. Güllü, "Statistical process control in machining, a case study for machine tool capability and process capability," *Materials & Design*, vol. 27, no. 5, pp. 364-372, 2006/01/01/ 2006, doi: https://doi.org/10.1016/j.matdes.2004.11.003.
[34] E. V. Gijo, "Improving Process Capability of Manufacturing Process by Application of Statistical Techniques," *Quality Engineering*, vol. 17, no. 2, pp. 309-315, 2005/04/01 2005, doi: 10.1081/QEN-200056494.
[35] J. S. Oakland, "Chapter 10 - Process capability for variables and its measurement," in Statistical Process Control (Sixth Edition), J. S. Oakland Ed. Oxford: Butterworth-Heinemann, 2008, pp. 257-273.
[36] M. Attaran, "Why does reengineering fail? A practical guide for successful implementation," *Journal of management development*, vol. 19, no. 9, pp. 794-801, 2000.

[37] A. Fretheim and O. Tomic, "Statistical process control and interrupted time series: a golden opportunity for impact evaluation in quality improvement," BMJ Quality & Safety, vol. 24, no. 12, pp. 748-752, 2015, doi: 10.1136/bmjqs-2014-003756.
[38] N. P. Grigg, J. Daly, and M. Stewart, "Case study: the use of statistical process control in fish product packaging," *Food Control*, vol. 9, no. 5, pp. 289-297, 1998/10/01/ 1998, doi: https://doi.org/10.1016/S0956-7135(98)00018-8.
[39] E. M. Saniga, "Economic Statistical Control-Chart Designs With an Application to and R Charts," *Technometrics*, vol. 31, no. 3, pp. 313-320, 1989/08/01 1989, doi: 10.1080/00401706.1989.10488554.
[40] A. Mostafaeipour, "The use of Statistical Process Control Technique in Ceramic Tile Manufacturing: Case Study," *International Journal of Applied Information Systems (IJAIS)*, vol. 2 No 5, no. ISSN: 2249 0868, 2012.
[41] Y. M. Awaj, A. P. Singh, and W. Y. Amedie, "Quality improvement using statistical process control tools in glass bottle manufacturing company," 2013.
[42] D. K. Ved Parkash, Rakesh Rajoria, "Statisitcal Process Control " *International Journal of Research in Engineering and Technology*, vol. 2, no. 08, 2013.
[43] V. Vashisht. "An Investigation into the Various Statistical Process Control Tools."
[44] S. B. T. Vaibhav Sonje, S.R Pande, Girish Katkar, "Statistical Porcess Control Implementation for Process Optimization and Better Quality," *Proceeding of Academicsera 20th International Conference*, Montreal, Canada, 2018.
[45] D. A. Tegegne, D. Kitaw, and B. Eshetie, "Advances in statistical quality control chart techniques and their limitations to cement industry," (in English), Cogent Engineering, vol. 9, no. 1, Jan 2022

2023-11-22 2022, doi: https://doi.org/10.1080/23311916.2022.2088463.
[46] P. Qiu, "Big Data? Statistical Process Control Can Help!," The American Statistician, vol. 74, no. 4, pp. 329-344, 2020/10/01 2020, doi: 10.1080/00031305.2019.1700163.
[47] G. Pang, C. Shen, L. Cao, and A. V. D. Hengel, "Deep Learning for Anomaly Detection: A Review," ACM Comput. Surv., vol. 54, no. 2, p. Article 38, 2021, doi: 10.1145/3439950.
[48] V. Chandola and V. Kumar, "Outlier Detection : A Survey," ACM Computing Surveys, vol. 41, 01/01 2009.
[49] A. A. Cook, G. Mısırlı, and Z. Fan, "Anomaly Detection for IoT Time-Series Data: A Survey," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6481-6494, 2020, doi: 10.1109/JIOT.2019.2958185.
[50] A. Nassif, M. Abu Talib, Q. Nasir, and F. Dakalbab, "Machine Learning for Anomaly Detection: A Systematic Review," IEEE Access, vol. PP, pp. 1-1, 05/24 2021, doi: 10.1109/ACCESS.2021.3083060.
[51] A. A. Ghorbani, W. Lu, and M. Tavallaee, "Theoretical Foundation of Detection," in Network Intrusion Detection and Prevention: *Concepts and Techniques*, A. A. Ghorbani, W. Lu, and M. Tavallaee Eds. Boston, MA: Springer US, 2010, pp. 73-114.
[52] I. Aichouri, A. Hani, N. Bougherira, L. Djabri, H. Chaffai, and S. Lallahem, "River Flow Model Using Artificial Neural Networks," *Energy Procedia*, vol. 74, pp. 1007-1014, 2015/08/01/ 2015, doi: https://doi.org/10.1016/j.egypro.2015.07.832.
[53] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 157-166, 1994, doi: 10.1109/72.279181.
[54] S. Manocha and M. A. Girolami, "An empirical analysis of the probabilistic K-nearest neighbour classifier," *Pattern Recognition Letters*, vol. 28, no. 13, pp. 1818-1824, 2007/10/01/ 2007, doi: https://doi.org/10.1016/j.patrec.2007.05.018.

[55] V. Vapnik, "Statistical learning theory," 1998.
[56] S. Hochreiter, "The Vanishing Gradient Problem During Learning Recurrent Neural Nets and Problem Solutions," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 6, pp. 107-116, 04/01 1998, doi: 10.1142/S0218488598000094.
[57] P. J. Werbos, "Generalization of backpropagation with application to a recurrent gas market model," *Neural Networks*, vol. 1, no. 4, pp. 339-356, 1988/01/01/ 1988, doi: https://doi.org/10.1016/0893-6080(88)90007-X.
[58] X. H. Le, H. Ho, G. Lee, and S. Jung, "Application of Long Short-Term Memory (LSTM) Neural Network for Flood Forecasting," Water, vol. 11, p. 1387, 07/05 2019, doi: 10.3390/w11071387.
[59] X.-H. Le, H. V. Ho, G. Lee, and S. Jung, "Application of Long Short-Term Memory (LSTM) Neural Network for Flood Forecasting," Water, vol. 11, no. 7, p. 1387, 2019. [Online]. Available: https://www.mdpi.com/2073-4441/11/7/1387.
[60] T. J. Lee, J. E. Gottschlich, N. Tatbul, E. Metcalf, and S. B. Zdonik, "Greenhouse: A Zero-Positive Machine Learning System for Time-Series Anomaly Detection," vol. abs/1801.03168, 2018.
[61] M. E. Villa-Pérez, M. Á. Álvarez-Carmona, O. Loyola-González, M. A. Medina-Pérez, J. C. Velazco-Rossell, and K.-K. R. Choo, "Semi-supervised anomaly detection algorithms: A comparative summary and future research directions," *Knowledge-Based Systems*, vol. 218, p. 106878, 2021/04/22/ 2021, doi: https://doi.org/10.1016/j.knosys.2021.106878.
[62] S. Omar, A. Ngadi, and H. H. Jebur, "Machine learning techniques for anomaly detection: an overview," *International Journal of Computer Applications*, vol. 79, no. 2, 2013.
[63] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "LOF: identifying density-based local outliers," presented at the Proceedings of the 2000 ACM SIGMOD international conference on Management of data, Dallas, Texas, USA, 2000. [Online]. Available: https://doi.org/10.1145/342009.335388.
[64] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," ACM Comput. Surv., vol. 41, no. 3, p. Article 15, 2009, doi: 10.1145/1541880.1541882.
[65] C. M. Ahmed, G. R. M. R, and A. P. Mathur, "Challenges in Machine Learning based approaches for Real-Time Anomaly Detection in Industrial Control Systems," presented at the Proceedings of the 6th ACM on Cyber-Physical System Security Workshop, Taipei, Taiwan, 2020. [Online]. Available: https://doi.org/10.1145/3384941.3409588.
[66] P. H. Tran, A. Ahmadi Nadi, T. H. Nguyen, K. D. Tran, and K. P. Tran, "Application of machine learning in statistical process control charts: A survey and perspective," in Control charts and machine learning for anomaly detection in manufacturing: Springer, 2022, pp. 7-42.
[67] T. Rato, M. Reis, E. Schmitt, M. Hubert, and B. De Ketelaere, "A systematic comparison of PCA-based statistical process monitoring methods for high-dimensional, time-dependent processes," AIChE Journal, vol. 62, no. 5, pp. 1478-1493, 2016.
[68] R. L. Mason and J. C. Young, "Multivariate statistical process control with industrial applications." SIAM, 2002.
[69] H. D. Nguyen, P. H. Tran, T. H. Do, and K. P. Tran, "Quality Control for Smart Manufacturing in Industry 5.0," in Artificial Intelligence for Smart Manufacturing: Methods, Applications, and Challenges: Springer, 2023, pp. 35-64.
[70] Á. L. P. Gómez, L. F. Maimó, F. J. G. Clemente, J. A. M. Morales, A. H. Celdrán, and G. Bovet, "A methodology for evaluating the robustness of anomaly detectors to

adversarial attacks in industrial scenarios," *Ieee Access*, vol. 10, pp. 124582-124594, 2022.
[71] R. Jiang, Y. Xue, and D. Zou, "Interpretability-aware industrial anomaly detection using autoencoders," *IEEE Access*, 2023.
[72] F. Bachinger, G. Kronberger, and M. Affenzeller, "Continuous improvement and adaptation of predictive models in smart manufacturing and model management," *IET Collaborative Intelligent Manufacturing*, vol. 3, no. 1, pp. 48-63, 2021.
[73] U. M. Mbanaso, L. Abrahams, and K. C. Okafor, "Research Philosophy, Design and Methodology," in Research Techniques for Computer Science, Information Systems and Cybersecurity, Cham: Springer Nature Switzerland, 2023, pp. 81-113.
[74] S. Hunziker and M. Blankenagel, "Introducing Research Designs," in Research Design in Business and Management: A Practical Guide for Students and Researchers, Wiesbaden: Springer Fachmedien Wiesbaden, 2024, pp. 1-17.
[75] C. L. Kimberlin and A. G. Winterstein, "Validity and reliability of measurement instruments used in research," (in eng), Am J Health Syst Pharm, vol. 65, no. 23, pp. 2276-84, Dec 1 2008, doi: 10.2146/ajhp070364.
[76] A. N. Vincent, K. Sakthidasan, A. Gadekar, and U. Bagurubumwe, "Machine Learning and Deep Learning Techniques for Predictive Modeling of Marine Ecosystem – A case of Flic en Flac Region, Mauritius," in 2023 *Second International Conference On Smart Technologies For Smart Nation* (SmartTechCon), 18-19 Aug. 2023 2023, pp. 1329-1335, doi: 10.1109/SmartTechCon57526.2023.10391505.
[77] S. M. Rasdas, A. C. Fajardo, and J. S. Limbago, "Predicting Abundance of Fish Species Populations in Manila Bay, Philippines Based on Ensemble Learning Approach," in 2023 20th International Joint Conference on Computer Science and Software Engineering (JCSSE), 28 June-1 July 2023 2023, pp. 129-134, doi: 10.1109/JCSSE58229.2023.10201953.
[78] M. Ospici, K. Sys, and S. Guegan-Marat, "Prediction of Fish Location by Combining Fisheries Data and Sea Bottom Temperature Forecasting," Cham, 2022: *Springer International Publishing, in Image Analysis and Processing* – ICIAP 2022, pp. 437-448.
[79] L. Lønmo and G. Muller, "7.1.2 Concept Selection - Applying Pugh Matrices in the Subsea Processing Domain," *INCOSE International Symposium*, vol. 24, no. 1, pp. 583-598, 2014, doi: https://doi.org/10.1002/j.2334-5837.2014.tb03169.x.
[80] D. Maulud and A. M. Abdulazeez, "A Review on Linear Regression Comprehensive in Machine Learning," *Journal of Applied Science and Technology Trends*, vol. 1, no. 2, pp. 140-147, 12/31 2020, doi: 10.38094/jastt1457.
[81] N. Desai and V. Patel, "Linear Decision Tree Regressor: Decision Tree Regressor Combined with Linear Regressor,"July 2021.
[82] Z. Jun, "The Development and Application of Support Vector Machine," *Journal of Physics: Conference Series*, vol. 1748, no. 5, p. 052006, 2021/01/01 2021, doi: 10.1088/1742-6596/1748/5/052006.
[83] T. Joachims, "Making large scale SVM learning practical," Technical reports, 1998.
[84] C. Cortes and V. Vapnik, "Support-vector networks," Machine Learning, vol. 20, no. 3, pp. 273-297, 1995/09/01 1995, doi: 10.1007/BF00994018.
[85] J. H. Friedman, "Stochastic gradient boosting," *Computational Statistics & Data Analysis*, vol. 38, no. 4, pp. 367-378, 2002/02/28/ 2002, doi: https://doi.org/10.1016/S0167-9473(01)00065-2.
[86] J. Friedman, T. Hastie, and R. Tibshirani, "Special Invited Paper. Additive Logistic Regression: A Statistical View of Boosting," The Annals of Statistics, vol. 28, no. 2, pp. 337-374, 2000. [Online]. Available: http://www.jstor.org/stable/2674028.

[87] A. Natekin and A. Knoll, "Gradient boosting machines, a tutorial," *Frontiers in neurorobotics*, vol. 7, p. 21, 2013.
[88] D. Chicco, M. J. Warrens, and G. Jurman, "The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation," PeerJ Computer Science, vol. 7, p. e623, 2021/07/05 2021, doi: 10.7717/peerj-cs.623.