IMPROVING SEAFOOD PRODUCTION THROUGH DATA SCIENCE METHODS

BAHAREH TEIMOURI LOTFABADI

A THESIS SUBMITTED TO
THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE
OF
MASTER OF ARTS

GRADUATE PROGRAM IN INFORMATION SYSTEMS AND
TECHNOLOGY
YORK UNIVERSITY
TORONTO, ONTARIO

August 2021

Abstract

Global production of seafood has quadrupled over the past 50 years. Seafood production is characterized by one of the highest waste rates in the food industry reaching up to 50% of the original raw material. Therefore, seafood companies are interested in reducing their waste rates, thus increasing production yields. In this thesis, we apply a Data Science (DS) methodology and suggest an extended DS framework to address theoretical and practical issues in the seafood industry. The framework encapsulates data processing, statistical, machine learning, visualization and optimization capabilities. The research employs unique real-world data collected in a seafood production facility over a 2-year period. The study will contribute to the economic well-being of the individual seafood producers as they could perform their business planning and forecasting in a more informed and predictive way as well as to the overall sustainability of the seafood industry due to the waste rate reduction.

Keywords: Seafood Production, Waste Rate, Production Yield, Data Science, Machine Learning, Framework

Acknowledgement

I would like to sincerely acknowledge all the great people for their unique help and support in realization of this study. First and foremost, I would like to thank my research supervisor, Prof. Peter Khaiter from the School of Information Technology, for offering me a great opportunity to conduct this study under his expert supervision and guidance throughout my master's education at York University. I am grateful to Mr. Eric Enno Tamm, CEO and Co-Founder of the ThisFish, Inc., for his great cooperation in terms of sharing his invaluable expertise and insights into seafood production industry and providing access to the datasets underlying this project. This research was supported by the National Research Council of Canada Industrial Research Assistance Program (NRC IRAP) Grant No. 2019-924246 and the Mathematics of Information Technology and Complex Systems (MITACS NCE) Grant No. IT19694-FR57022.

I would like to greatly appreciate the members of my Thesis Examination Committee, Prof. Xiaohui Yu from the School of Information Technology and Prof. Berta Esteve-Volart from the Department of Economics, for dedicating their valuable time, professional expertise, and attention to serving on the Committee, reviewing this thesis, and providing valuable comments and recommendations.

I would like to thank the administrative staff of the School of Information Technology and MAIST Graduate Programme, Ms. Ellis Lau, and Ms. Ummay Efendi, for their great help and support throughout various administrative steps of the process. Finally, lots of gratitude goes to my dear and beloved husband, Hamed, who has been always solidly and whole heartedly supporting me, personally and professionally.

Table of Contents

List of Figures

List of Tables

Chapter 1. Introduction

Consumption of seafood has increased over the last decade, and, thus, seafood production has become nowadays one of the most important industries in the world. Globally, more than 155 million tonnes of seafood are being currently produced every year. In fact, many countries have large seafood production facilities. Among the world's leaders are the United States, Japan, China, Russia, South Korea, Peru and India. Seafood production is quite lucrative industry generating annually billions of dollars of profit (Ritchie et al., 2019).

At the same time, seafood production is characterized by one of the highest rates of waste in the food industry, reaching sometimes as high as 50% of the original raw material. Of this amount, seafood processing alone is responsible for almost a half of the waste volumes which is obviously not only one of the serious challenges for the sector due to the tangible economic losses being incurred (Himmetoglu, 2017), but also a serious negative impact on the environmental conditions (Arvanitoyannis et al., 2008).

There are three groups of factors in seafood production: (1) natural variables (e.g., fish population, size, sexual maturity, *etc*.); (2) raw material parameters (e.g., harvest area, harvest method, treatment aboard a vessel, *etc*.); and (3) process control parameters (e.g., time and temperature of thawing and cooking), which affect fish quality as well as production yields and waste volumes. Out of these three types of variables and, unlike terrestrial agriculture, seafood companies cannot immediately control the natural variables. Therefore, companies are trying to maximize their production yields (i.e., a ratio of the final product to the raw material used) by focusing on the two other types of factors: raw material parameters, e.g., to match fish quality with product type and process control parameters, e.g., to optimize the time and temperature of thawing and cooking in the case of canned fish production.

Motivated by the above considerations, in this research, we study the effects of various raw material and process control parameters on production yields, as well as the capability of predicting the production yields based on these two sets of parameters. The latter task is aimed at helping the seafood producers to maximize their production yields and reduce the waste rates which will have not only positive sustainable effect but also, from the economic standpoint, would save hundreds of thousands or even millions of dollars a year depending on the seafood processor scales (Himmetoglu, 2017; Tamm, 2020).

This research is based on real-world data which has been electronically collected in a seafood production facility located in Bangkok, Thailand using a unique Tally software system developed by ThisFish Inc, over a period of 2 years from January 2018 until November 2020. In this thesis, we applied a Data Science (DS) approach as a methodological foundation of the study and suggested an extended DS framework to deal with the whole scope of the issues and questions.

In order to develop the DS framework, we addressed the following Research Questions (RQ):

RQ 1. To introduce the structural components of the DS framework, phases in its development lifecycle and a common roadmap to be followed in future projects in the seafood production domain.

RQ 2. To formalize the notion of, and formulate a mathematical expression to compute, the seafood Production Yield Values (*PYV*) based on available datasets.

RQ 3. To investigate the relationships between *PYV* and various Process Control parameters and assess in which way and to what extend they affect *PYV*.

RQ 4. To investigate the relationships between *PYV* and Raw Material parameters and assess in which way and to what extend they affect *PYV*.

RQ 5. To investigate different techniques for handling the categorical variables and recommend the most efficient encoding technique(s) suitable for seafood data with a view of their inclusion in the ML algorithms?

RQ 6. To analyze various ML algorithms in order to design a predictive ML modeling component of the framework.

RQ 7. To test and compare these algorithms using their relevant performance metrics to determine which algorithm(s) produce the best prediction results.

RQ 8. To study whether a Stacked Modeling (SM) architecture can improve the predictive performance of the ML models.

This research contributes to the body of knowledge in the field of DS and ML applications in the domain of seafood production industry. The novelty of this research is that a common methodology is suggested and elaborated and also given the lack of electronically collected datasets, to the best of our knowledge, there aren't prior comparable studies reported in the literature. ThisFish Inc. is the first company which invented a unique Tally software system and

successfully collected such suitable datasets, thus, enabling comprehensive analytical and modeling studies conducted in this project.

Structure of this thesis is as follows. Chapter 2 is dedicated to the review of the available relevant literature in the field of study. Chapter 3 elaborates on the methodology as well as methods, models and tools applied in this study. The DS framework, its components, development lifecycle and roadmap are also introduced in this Chapter. Chapter 4 deals with the Collection, Extraction, Transformation and Loading (CETL) tasks applied to the datasets in order to prepare them for the analytical and modeling steps. Chapter 5 presents all types of statistical analysis performed on the datasets to study the data distribution patterns and test hypotheses underlying relationships amongst features of the dataset. Chapter 6 presents ML predictive modeling component of the framework. Here, various ML algorithms are analyzed and compared to determine which algorithm(s) produce the best prediction results. The advantages of a Stacked Modeling (SM) architecture to improve the predictive performance of the ML models are also investigated in this Chapter. Finally, Chapter 7 discusses the optimization module of the DS framework and its implementation as directions of future work and concludes the thesis.

Chapter 2. Literature Review

There has always been an idea in various fields such as design, business, medicine, or manufacturing of finding relevant patterns in their collected data. This task has been traditionally approached through statistical methods meant to understand hidden relationships within large volumes of data. Recently, it is complemented by new theories, methods and algorithms developed in an ever-increasingly emerging areas of artificial/computational intelligence, namely, data science and machine learning. Data science (DS) comprises a mixture of approaches and methods from data management, statistics, artificial intelligence as well as machine learning. Researchers and analysts from various domains including retail, insurance, engineering, fraud detection and marketing have been applying these methods in their studies to better understand in-hand data and its underlying relationships (Harding et al., 2006).

Data science and machine learning form an expanding field of science given the growing volumes of available data and the number of applications from across many disciplines relying on these techniques. Historically, they had been applied in the engineering field since 1990s (Lee, 1993; Irani et al., 1993; Piatetsky-Shapiro, 1999) for better understanding of their data targeting to improve many tasks like quality assurance, decision support, scheduling, production, design, predictive maintenance, and fault detection and diagnosis. Primarily, data analysis helps in extracting and recognizing hidden patterns in the important parameters of engineering control processes. Another major benefit of the new methods is that the data analysis task could be performed in real-time regime during the operation of the manufacturing process in question (Harding et al., 2006; Wrobel et al., 2019).

Infrastructure of machine learning tools has been increasingly improved for performing a better generation, testing and refinement of scientific models. These models can be tailored for addressing extremely complex phenomena. Importantly is that not only the researchers from within hardcore artificial intelligence or statistics contribute to maturation of machine learning methods, but also significant contributions are being made by the practitioners in particular fields applying these methods in their respective problem domains (Butler et al., 2018).

One of the fields in which DS and ML modeling frameworks have become very helpful is the healthcare industry. Nowadays, DS techniques are contributing to this field mostly by streamlining the administrative procedures of hospitals and health centres. As well, ML models are being used in this field by predicting the mapping and treatment of infectious diseases and

personalized medical procedures (Shayea et al., 2011; Burbidge et al., 2001; Altschul et al., 1997; O'Driscoll et al., 2013). In this way and to shrink the expenses of Electronic Medical Records (EMR), data scientists help developing various software based on ML predictive algorithms. Similarly, applying ML strategies, has helped doctors in predicting illness and thus in recommending better treatment and modeling the progression of certain class of diseases (Koh et al., 2005; Reddy et al., 2015).

The healthcare industry has benefitted a lot from ML predictive strategies for enhancing its clinical decision making processes and developing guidance tools for diagnoses and treatment possibilities for improving efficiency and efficacy of caregiving tasks and thus higher level of patient satisfaction (Kohn et al., 2014; Lutz et al., 2021).

Additionally, pathologists utilize ML framework for performing faster and even more reliable diagnoses as well as identification of new types of therapies and/or treatments more beneficial to their patients. ML algorithms have helped improving speed of diagnosis of breast cancer and in planning of radio-pathetical and surgical procedures, e.g., correctly differentiating between healthy organs and tumors and its application in research in fight against cancer by immune system of the human body, specifically in the field of immune-oncology (Yue et al., 2018; Thomas, 2020).

ML techniques are also used in space exploration missions (Krishna et al., 2021). This latter is mostly due to the large volume of data that is produced throughout every space mission. As such, various data warehousing techniques have been developed by data scientists for better collection of such amount of data. This is then recalled in its later use in prediction tasks for performing more successful execution of space exploration missions.

Energy production and management industries have applied ML methods for their forecasting needs to design a more accurate response to demand and supply (Ghoddusi et al., 2019; Sorensen et al., 2018; Riad et al., 2010; Assarzadeh et al., 2008; Huang et al., 2015).

Another domain in business where ML methods have been widely applied is in supporting decision making procedures based on reliable and accurate predictions generated by ML algorithms (Cavalcante et al., 2016; Linden et al., 2003; Xie et al., 2012; Ren et al., 2018; Weimer et al., 2016). Within agriculture industry, decision support techniques developed based on ML methodologies encompass climate as well as available water and energy resources along with other potentially case-specific parameters for helping farmers and fisheries make better decisions for

their future planning activities (Liakos et al., 2018). Decision support systems also help business leads predict trends and thus anticipate changes in their income and expense factors for better identification of problems and expediting more confident decision makings. DS tools, like data visualization tools, also help better presentation of business data to the clients and/or executives (Mohd Ali et al., 2020).

The so-called Customer Recommendation Engines have also been using a variety of ML techniques for improving the customer service and thus to offer better experiences (Ballestar et al., 2018). In such a context, data points get processed exclusively per every individual client, including past purchases made by the client and other databases, such as current inventory and the demographic and revenue trends of the business, in order to recommend relevant products and services which would be more appealing to every client.

For instance, big e-commerce businesses such as Walmart or Amazon heavily depend on such recommendation engines for further personalization and thus expediting the shopping experience of their clients. Another famous user of these ML technologies is the entertainment business, Netflix, which delivers very personalized recommendations to its viewers based on viewing history of every viewer on Netflix and other entertainment media, and other potentially accessible info on viewer's favourite media products. Similarly, online video website, YouTube, applied ML recommendation strategy for assisting its clients for accessing their potentially more favoured media within a lesser time (Saleem et al., 2021).

Various industries apply DS and ML techniques to predict and anticipate the time, for when their relationship with a specific client shall begin to deteriorate, and thus to find a solution for its improvement. In fact, this is one of the historical problems emerging in various industries on how to best address their so-called customer churn (Vafeiadis et al., 2015). For this purpose, DS techniques retrieve patterns from the large volumes of data on various types, such as sales, historical and demographic data, for better identification and understanding of root causes and main reasons that an industry loses its clients.

DS techniques and ML modeling capabilities have made major contributions to the manufacturing field. These contributions include, but are certainly not limited to, shrinking losses, for instance, production waste and yield metrics, production quality and throughput. Also, ML techniques have successfully increased manufacturing capacity factor through prediction and thus optimization of the production procedures. On the other page, ML methods have empowered

procedures with a tool of making best decisions in terms of expansion and growth of their manufacturing lines in the most beneficial way using reliable forecasts of their future production rates (Onsree et al., 2021; Kanga et al., 2020; Razaviarab et al., 2019).

The other area, in which DS analytical tools and ML prediction capabilities have contributed to the production field, is reducing maintenance costs by providing reliable predictions on various maintenance needs of the equipment which would enable appropriate pre-caution measures. This latter also results in lowering labour costs and, in turn, reduction of quantity of inventory and material waste (Wigley et al., 2016; Min et al., 2019; Weichert et al., 2019).

Predicting Remaining Useful Life (RUL), based on which manufacturers understand machines and equipment behaviour, aims to increase production yield as well as provide a better maintenance planning and higher equipment and machine health. This latter prediction task has been widely taken care of by ML algorithms and DS tools. Another important aspect of RUL prediction is its impact in decreasing so-called "unpleasant surprises" in manufacturing context resulting directly in less unforeseen downtime of production lines (Ren et al., 2017; Carroll et al., 2018).

ML also helps manufacturers in supply chain management by providing reliable forecast of their inventory needs and thus better planning and management of the manufacturing tools and parts resulting in more accurate and synchronized monitoring of production flows. As a result, quality control will become also better actionable with less production deficiencies (Kanawaday et al., 2017; Carvalho et al., 2019).

Learning Human-Robot interaction, especially in production environments where workers and machines or equipment need to work together, will certainly contribute to improving overall production efficiency as well as workers' safety. Demand forecasting and sales prediction, which are two areas where ML has matured a lot, will also help manufacturing and production executives respond quickly and efficiently to the changes in market to better address clients' needs and increase their economic benefits (Wenzel et al., 2019; Carbonneau et al., 2008; Ni et al., 2020; Subramanian, 2020).

Another domain where ML and its subfield called Deep Learning (DL) is helping many industries, in particular, manufacturing and production, is image analysis and image processing. This latter, for instance, helps in automation of quality control of manufactured products without

human intervention and based on application of various ML- and DL-based image processing algorithms, where damaged products get classified out of undamaged ones (Park et al., 2017).

Additionally, there are other areas where manufacturing companies benefit from image recognition and classification techniques: (1) using robots equipped with computer vision and ML capabilities to monitor shelves and thus to report which specific tool may be out-of-stick or is very low in its quantity; (2) applying image recognition techniques to make sure that the products have been correctly and fully scanned prior to their shipping, so to shrink loss probabilities; (3) monitoring safety of production lines by applying image analysis techniques for automated classification of dangerous situations, like outbreak of a machine, or suspicious activities, like unauthorized or incorrect use of a machine by a worker, which would endanger the worker or violate workplace safety protocols (Peng et al., 2021).

The next area of applications where industries are using ML methods is administering their financial transactions, business procedures and software development (Appiahene et al., 2020). This latter includes, in particular, manufacturing and production procedures and systems (Jaensch et al., 2018), enterprise financial organizations (Luo, 2021) as well as software development and testing (Wan et al., 2019).

In fact, ML-based technologies have been widely applied for improving efficiency within many industries. Some examples of this include: (1) many production companies, financial enterprises and consulting firms which utilize ML for expediting their work routines as well as for reducing potential human errors (Eshwein, 2019); (2) operations teams within every production organization which use various ML- or DL-based technological solutions for monitoring production lines, equipment and workers, for recognizing product deficiencies, safety violations, machine malfunctioning, workers violation of safety routines, and thus shrinking the likelihood of any unanticipated or unplanned issues and disruptions in production line and final products (Demir et al., 2019); (3) Information Technology (IT) departments which use or develop ML pipelines for automation of their software validation tasks which would, in turn, result in significant escalation in speed and improvement in accuracy and precision of such tasks leading to reduced costs as well as better software products (Biessmann et al., 2021).

Data scientists have as well widely applied ML algorithms for performing Information Retrieval (IR) tasks. In particular, recently developed Natural Language Processing (NLP) algorithms are capable of automatic recognition of important textual information out of a collection

of documents regardless of their nature, being represented by structured, semi-structured or unstructured datasets.

In fact, using ML methods for better and faster understanding of documents has become appealing to many industries. For instance, organizations may apply these techniques for processing their textual data, including retrieving info from tax documents or bonding contracts by improving the already in-place human-based way of performing these tasks. Production and manufacturing industries are also not an exception as they need to extract certain info, like manufacturing standards applicable to a particular product out of thousands of pages documents very quickly and reliably (Lavelli et al., 2008).

Environmental field as well has been benefitting from DS techniques and ML-based technologies. Recent research within ML and DL fields focusing on Environment Water Management (EWM) has helped in addressing remote sensing problems, such as semantic segmentation, object tracking, image recognition and processing techniques, scene classification and object detection (Maganathan et al., 2020).

Along these lines, DS and ML have contributed significantly to the Environmental Remote Sensing (ERS) domain. The core idea in ERS is using aircrafts or satellites for aggregating data about particles and areas. Thus, DS statistical tools and ML modeling techniques have been helping this field by analyzing such collected large datasets generated day-by-day and thus for monitoring earth condition or forecasting its future (Chang et al., 2017).

Soil science is another environmental field which studies soil development, its taxonomy or ecology. The main venues where ML has been contributing to this domain is in monitoring and forecasting the soil health including its resistance and nutrition status which then is significantly important in foreseeing and thus accosting for flood or drought conditions in the future horizon (Padarian et al., 2020; Fischetti et al., 2019; Donida et al., 2016; Cuadra et al., 2016; Khmaissia et al., 2018; Lakshmanan et al., 2015).

At the same time, we have found a very narrow DS and ML literature directly related to the seafood production field. This latter can be explained by the fact that, to the best of our knowledge, no electronically collected datasets of seafood production process have ever been available prior to this project, thus, limiting or even precluding a systematic and full-scale DS-based analytics and ML-based modeling.

Nevertheless, there are a few studies in the seafood production field which have done some modeling tasks related to the seafood production but not based on ML algorithms. For instance, provided that precooking time-duration of raw fish is a vital part of the tuna manufacturing process, (Debeer et al., 2015) describes manufacturing processes for precooking tuna fish which use a conventional atmospheric pre-cooker, where fish thickness as well as fish weights, are measured to determine variation of thickness based on the size of fish. Then, a simulation-based model using finite difference techniques is used to conduct a study on the impact of initial backbone temperatures, size of fish as well as ambient steam temperatures on precooking times. At least, one main shortage of this latter study is that it only accounts for pre-cooking time-duration parameter whilst many other factors impacting the seafood production process have been left aside.

Also, Pérez-Martín et al. (1989) introduced a semi-empirical model of precooking process for albacore tuna fish species which have been collected and processed under strict experimental conditions. The prediction of the optimal precooking time has been then performed based on this latter model using a simulation framework. In addition, further detailed illustrations on the variations in muscle of tuna during these processes have been provided by Bell et al. (2001). One main challenge with this latter study is its applicability to other tuna fish production scenarios, e.g., considering for other species or less restrictive experimental conditions.

Another example in the seafood production field is the Hazard Analysis and Critical Control Point (HACCP) plans where the science-driven evidence is utilized to develop and verify a so-called preventative process control aiming to limit and bound a specific food hazard in an efficient and effective way (Adams et al., 2018; Administration, 2017). Important that the National Oceanic and Atmospheric Administration (NOAA), whose main duty is to manage the fisheries in the United States, reports the lack of reliable and comprehensive datasets as its major obstacle (Department, 2012). This latter fact would limit the capability of developing models and determining sustainable limits of catching. Recently, nevertheless, NOAA has been using ML techniques to better understand available fishery-related dataset and, mainly, to address the overfishing issue as the future comes (Schlossberg, 2018; Marranzino, 2018). However, these studies are more related to managing fishing activities and, thus, forecasting potential overfishing issues. Therefore, actual seafood production has not been accounted for in these studies.

The absence of systematic and full-scale DS-based studies of the seafood production has motivated the present research which is aimed to address said gap.

Chapter 3. Methodology

Data Science is a cross-disciplinary field of science which develops and applies scientific methods, algorithms and techniques to primarily extract knowledge and useful relevant insights based on a given dataset (Dhar, 2013). In this way, Data Science field aggregates vast variety of statistical, data analytical as well as informatics tools to analyze, understand and derive such insights and conclusions (Hey, 2009). Therefore, Data Science will employ a wide skillset spanning from graphics design and data visualization, statistics, information sciences and integration to complex computer sciences and systems and even communication and business (Loukides, 2020).

Every Data Science (DS) project essentially consists of a certain main set of steps starting from data collection, data cleansing, exploratory data analysis, modeling and ending with the actual deployment. A summary infographic of DS project lifecycle is shown in Figure 1.

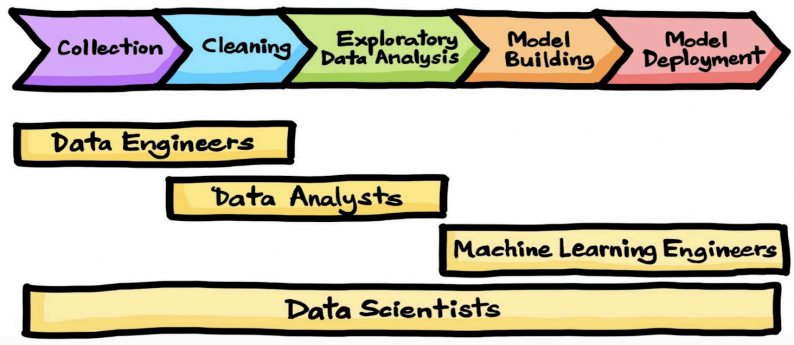


Figure 1 Main Steps of Data Science Project (Source: Loukides, 2020)

To the best of our knowledge, the Seafood Production is one of the applied fields, in which Data Science and its benefits have never been explored. As such, no intelligent computer-based solution was ever developed to reliably predict the production yield and reversely the waste rate in seafood production industry. In this research, DS approach entailing the above main steps has been developed and applied with the goal of notably and considerably contributing to the seafood production field by empowering it with reliable analytical and prediction capabilities. It is expected that the latter will also bear substantial economic benefits to this field in terms of increased production and reduced waste.

For this study, we suggest a novel DS framework as reflected in an improved and detailed data science roadmap shown in Figure 2. The framework starts with data collection on each technological stage, spanning from catching the raw Tuna fish all the way through to the final production of Tuna cans, followed by accessing and extracting corresponding data from the

databases accumulated by the industry companies. On the next step of the framework, various Data Preprocessing and Data Cleansing procedures are to be performed preparing data for Statistical Analysis aimed at a better understanding of the underlying relationships between features and deriving useful insights, such as correlations between features and their distribution patterns. Feature Engineering and Extraction phase is meant to arrange the dataset for the modeling stage. Investigation of performance abilities of various ML algorithms and selection of the best predictive model is the next step of the framework. To complete the scope of the framework, an optimization phase embeds the best performing model and optimizes for the best set of parameters resulting in highest *PYV*. The optimization module of the DS framework as well as perspectives of its implementation and deployment are discussed in Chapter 7.

Figure 2 Detailed Roadmap of the DS Framework

In the following sections, we present the key methods, techniques and models most suitable for the seafood production industry and applied in the proposed DS framework.

3.1 Data Analysis

The procedures for analyzing data to find useful insights and conclusive patterns to support decision-making processes and perform better ML modeling are the main tasks of Data Analysis (Kudyba, 2014). Indeed, it entails diverse disciplines, techniques, and methods used within specific areas of finance, business or science (Pruneau, 2017). Particularly in this research, we have applied two main categories of Data Analysis techniques, namely Testing for Normality and Parametric Statistical Tests.

Verifying the normal distribution is motivated by the fact that many of the statistical tests, such as correlation, regression, independent *t*-tests or ANOVA tests, are based on normality

assumption of the data. For this aim, Q-Q Plot (Glen, 2015; Makkonen, 2008), Histogram (Stangor, 2011; Jaradat et al., 2014), Boxplot (Praveen et al., 2017), and Heatmap (Wilkinson et al., 2012) as graphical methods and Kolmogorov-Smirnov (KS) test as a statistical method have been applied in the study.

Parametric Statistical Tests are methods for assessing correlation between the features and/or analyzing the distributions of features in the dataset. In this study, we have applied Pearson and Point-Biserial Correlation coefficient, independent *t*-test and ANOVA.

3.1.1 Kolmogorov-Smirnov (KS) Test

Named after its inventors, Andrey Kolmogorov and Nikolai Smirnov, the Kolmogorov-Smirnov (KS) method is a non-parametric test used very often in statistics to test the goodness of fit of one-dimensional and continuous empirical probability distribution against a reference (theoretical) probability distribution, e.g., normal distribution (Marsaglia et al., 2003; Vrbik, 2018; Oztuna et al., 2006).

3.1.2 Pearson Correlation Coefficient

The Pearson (or Pearson-Gamma) Correlation Coefficient method was suggested by Karl Pearson based on a relevant study conducted by Francis Galton in the 1880s. This was then followed by its mathematical articulation published by August Bravais in 1844 (Stigler, 1989).

Given a pair of random variables, (X, Y), the Pearson correlation coefficient, $\gamma_{X,Y}$ ranges from -1 to 1. (Holmes et al., 2020; Puth et al., 2014).

3.1.3 Point-Biserial Correlation

The point biserial (pb) correlation coefficient, denoted by r_{pb}, is a correlation coefficient utilized in studying the correlation between a continuous random variable (X) and a dichotomous categorical random variable (Y) (Linacre, 2008; Stigler, 1986).

To compute $\mathrm{r_{pb}}$, let us assume that the considered dichotomous variable, Y, takes two values: 0 and 1. Accordingly, the dataset is split into two groups: "group 1" that includes instances of X for which $Y = 1$ and "group 0" that includes the rest of instances of X for which $Y = 0$. Then, the point-biserial correlation coefficient is computed from Equation (1):

$$\mathrm{r_{pb}} = \frac{\mathrm{M_1 - M_0}}{\mathrm{s_n}} \sqrt{\frac{\mathrm{n_1 n_0}}{\mathrm{n^2}}}, \qquad (1)$$

where s_n denotes the standard deviation of X; M_1 is the mean value of X for the data points in "group 1"; M_0 is the mean value of X for the data points in "group 0"; n_1 denotes the number of data points in "group 1"; n_0 denotes the number of data points in "group 0"; n is the total number of data points in the dataset.

3.1.4 ANOVA Test

Analysis of Variance (ANOVA) provides a statistical test used to determine whether the mean values of two or more populations are equal. The first application of ANOVA was published in 1921 by Ronal Fisher (Fisher, 1921).

Null Hypothesis for ANOVA always states that there is no difference in the mean values while the Alternative Hypothesis is that the mean values are not equal (Gelman, 2005; Keppel et al., 1989).

3.1.5 Independent *t*-test

The Independent *t*- test is applied to analyze whether the mean values of two groups of data are equal in the statistical significance sense. The *t*-distribution was first derived in 1876 by Helmer and Lüroth (Pfanzagl et al., 1996). Independent *t*-test is applied in cases where the datasets being tested follow the normal distribution (Wendl, 2016).

There are two possibilities for defining Null Hypothesis of the Independent *t*-test: (1) *one-tailed* test which determines a particular sign of the difference between the mean values and (2) *two-tailed* test to verify the equality of the two mean values. (Guo et al., 2017).

3.2 Feature Engineering and Extraction

Feature extraction and engineering is the step in the DS lifecycle immediately preceding the application of ML predictive algorithms which begins with some initial set of measured independent features based on which further features are extracted (or engineered). The main purpose of so doing is to build new features which are more informative for subsequent predictive learning steps (Sarangi et al., 2020). In this study, we have mostly applied two feature engineering techniques, namely Encoding of Categorical Variables and Lasso Regression Coefficients.

3.2.1 Encoding of Categorical Variables

One type of variables, which in many applications affect the dependent variable, is so-called categorical (qualitative) variables of nominal (ordinal) scale (Alkharusi, 2012). However,

the majority of ML methods are designed to admit numerical inputs. Therefore, it is required to encode (or transform) such categorical variables into their numeric counterparts which would be admissible by the ML methods. This is done using certain encoding techniques (Carey, 2017; von Eye et al., 1996; Lantz, 2013; Myers et al., 2003; Education, 2017). These techniques are primarily Ordinal Encoding, One-Hot Encoding and Effect Encoding. In this study, One-Hot Encoding has been applied.

3.2.1.1 One-Hot Encoding

In "One-Hot encoding" method, the binary variables are introduced to represent the levels of categorical variable (Alkharusi, 2012; Carey, 2017). Every level of the categorical variable shall be encoded with 1 in its respective dummy variable's column and 0 for the remaining $(k-1)$ dummy variable's columns (Myers et al., 2003; Education, 2017; Ranganathan et al., 2019; Al-Sahaf et al., 2019). The technique can be expressed by Equation (2):

$$y_i = b_0 + \sum_{j=1}^{k} b_j \mathbb{1}_{ij} + e_i. \qquad (2)$$

In the above equation, y_i is the value of the target/independent variable at the i^{th} data instance; b_0 is the intercept value which is independent of the dummy variables; b_j is the coefficient of the j^{th} dummy variable; $\mathbb{1}_{ij}$ is the indicator function representing the j^{th} dummy variable at the i^{th} data instance, i.e., its value is 1 for the corresponding non-zero dummy variable at the i^{th} data instance and 0 for other dummy variables; e_i is the error of the prediction of y_i (Allen, 1997; Myers et al., 2003).

3.2.2 Lasso Regression Coefficient

Robert Tibshirani was first to formulate Least Absolute Shrinkage and Selection Operator (LASSO) in 1996 (Tibshirani, 1996). LASSO is a powerful technique which is very useful in two main applications: regularization and feature selection. From formulation standpoint, LASSO method imposes a constraint in terms of sum of the absolute values of the model parameters. This is accomplished by applying a shrinking, or so-called regularization process through penalizing the regression variables shrinking coefficients in such a way that some of them become zero. During feature selection process, the features with a non-zero LASSO regression coefficient are considered to be more important and thus will be selected for the model (Ranstam et al., 2018; Jacob et al., 2009).

3.3 Machine Learning Algorithms

Machine Learning (ML) as a part of AI domain is the study of computer-based algorithms whose prime objective is to build a model to learn from sample data and predict or decide based on newly available data, to improve prediction or decision accuracy in a timely manner without having been explicitly programmed for this purpose (Alpaydin, 2020). In every ML algorithm, the entire available dataset is split in two subsets. The first subset (training data) is used for the purpose of fitting the ML model. The second subset (test data) is used for the purpose of evaluating the performance of the trained model against the previously unseen data. This latter train-test splitting technique is used for performance evaluation of any supervised classification and regression algorithm.

ML algorithm can be classified under supervised, unsupervised, and semi-supervised approaches Amongst these approaches, supervised learning methods are being used in vast majority of research studies in the ML domain, which are also considered to be the most powerful methods (Butler et al., 2018; Kireeva et al., 2012). Depending on the data type and the goal of analysis there would be various learning methods available to be applied (see Table 1) (Agrawal et al., 2016).

Table 1 Most Commonly Used ML Algorithms for Supervised Approach

Modeling Technique	Capability	Brief Description
Naïve Bayes (John et al., 1995)	Classification	Bayes theorem-based probabilistic classifier
Bayesian Network (Bouckaert, 2004)	Classification	Describes probabilistic conditional relationships in between various variables on a graphical model
Logistic Regression (Hosmer et al., 1989)	Classification	Data is fitted into a sigmoid curve
Linear Regression (Weher et al., 1976)	Regression	Linear least square model is fitted based on the input dataset
k-Nearest Neighbour (Aha et al., 1991)	Classification and Regression	Makes predictions based on the most similar data points in training dataset
Artificial Neural Networks (Bishop, 1995; Fausett, 1994)	Classification and Regression	Uses stacked hidden neurons layers in between input and output features, where learning occurs by adjusting weights of network links
Support Vector Machines (Vapnik, 1995)	Classification and Regression	Draws hyperplanes in features space based on structural risk minimization techniques
Decision Table (Kohavi, 1995)	Classification and Regression	Builds rules containing various attributes combinations
Decision Stump (Witten et al., 2005)	Classification and Regression	Tree-based learning approach with only one level of decision tree
J48 (C4.5) Decision Tree (Quinlan, 1993)	Classification	Uses Gini impurity/ information gain to identify splitting rules of the tree
Alternating Decision Tree (Freund et al., 1999)	Classification	Alternative prediction and decision nodes are used in the tree, where a

		data instance goes through all potential paths
Logistic Model tree (Landwehr et al., 2005; Sumner et al., 2005)	Classification	Logistic regression is used at the leaves of the tree
M5 Model Tree (Wang et al., 1997; Quinlan, 1992)	Regression	Linear regression is used at the leaves of the tree
Random Tree	Classification and Regression	Randomly selected attributes subsets are considered
Reduced Error Pruning Tree (Witten et al., 2005)	Classification and Regression	Tree is constructed based on variance/information gain where, to prevent overfitting, reduced-error pruning technique is used for its pruning
Random Subspace (Ho, 1998)	Ensembling	Builds multiple trees by pseudo-randomly choosing features subsets in a systematic way
Random Forest (Breiman, 2001)	Ensembling	Constitutes ensemble of multiple trees

Resorting to the above table, we have chosen six most commonly used regression methods belonging to various types of ML approaches. These selected methods are Linear Regression as a Linear approach, Artificial Neural Network as a Deep Learning approach, Support Vector Machine as a Kernel-based approach, k-Nearest Neighbour as a Non-Parametric approach, Random Forest as a Bagging approach, AdaBoost as a Boosting approach. In this section, we review these ML algorithms.

3.3.1 Linear Regression

Amongst linear approaches in ML, Linear regression has been very rigorously studied and extensively applied to address various practical applications (Yan et al., 2009; Rencher et al., 2012). Linear Regression is an approach used in statistics to model the linear relationship between dependent variable and independent variables. The technique is further classified into simple linear regression if there is only one independent variable and multiple linear regression if there are more than one independent variable (Freedman, 2009).

3.3.2 Support Vector Machine

In the context of ML, Support Vector Machines (SVM) is a kernel-based learning model used for performing supervised learning. SVM modeling framework can support both classification as well as regression applications. By their construction, SVMs are counted amongst robust class of learning models. This is because SVMs are statistical learning methods based on the so-called Vapnik and Chervonenkis (VC) theory (Cortes et al., 1995; Ben-Hur et al., 2001). SVM performs mapping on the training data instances to certain points in the space in such a way

that the width of gap between the two considered classes be maximized (Hastie et al., 2008;Press et al., 2007). This is illustrated in Figure 3.

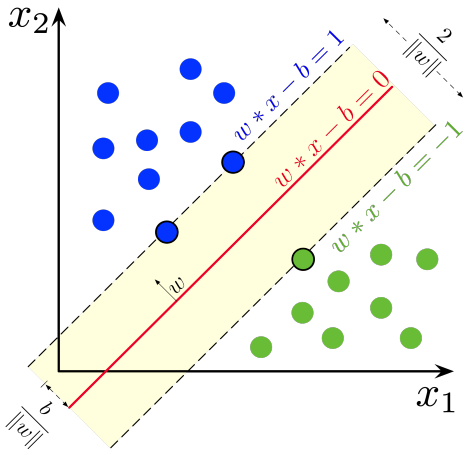


Figure 3 Support Vector Machine (SVM), Hyperplane and Main Idea for Classification (Source: Hastie et al., 2008)

3.3.3 Artificial Neural Networks

Artificial Neural Networks (ANNs) are one of the deep learning methods inspired by their biological neural network counterpart (Chen et al., 2019; da Silva et al., 2017). The general model of ANN is shown in Figure 4.

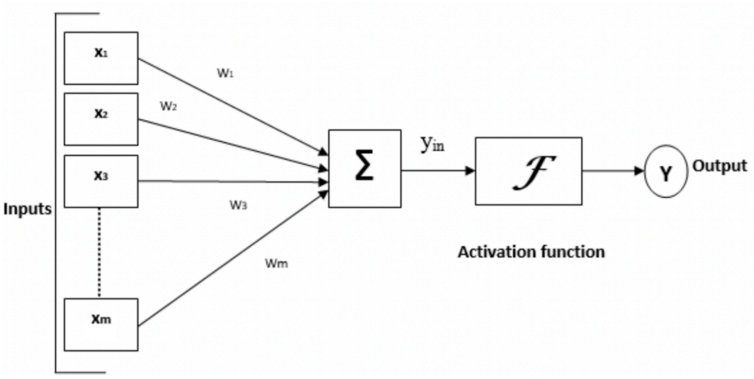


Figure 4 General Model of ANN (Source: (da Silva et al., 2017))

Learning in ANN model means finding the best weights for the links between the neurons of a specific network. This is done by assigning the input vector to the model, finding the outputted value, then comparing it with the actual desired output where the difference between the two values is called an error. In case of an error, the weights of the ANN will be adjusted until this error becomes sufficiently small.

3.3.4 k-Nearest Neighbour

k-Nearest Neighbours (k-NN) algorithm is a non-parametric method developed in 1951 and applied in both regression and classification tasks (Altman, 1992).

k-NN classification algorithm determines the class (amongst classes of the target feature) to which a given new data instance belongs. This latter is computed based on the majority of the classes of the k nearest neighbours amongst all the data instances of the training dataset of this new data instance being classified. Therefore, the class of the new data instance is set to be this majority class of the k neighbours located nearest to the new data instance (Piryonesi et al., 2020; Jaskowiak et al., 2011). See Figure 5.

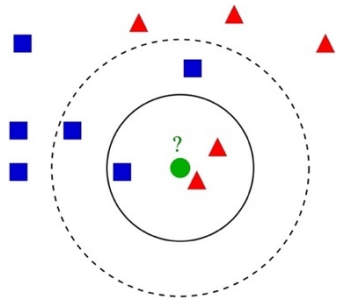


Figure 5 Implementation Example of k-NN Algorithm (Source: Jaskowiak et al., 2011)

3.3.5 Random Forest

Random Forests, also known as Random Decision Forests, are one of the ensemble bagging ML algorithms for performing classification as well as regression tasks (see Figure 6). The main idea behind random forest algorithms is to construct multiple decision trees during the training phase of the algorithm and thus to output either the majority of classes or the mean value of the predicted target values produced by every individual decision tree within the random forest as the class or target value prediction of the whole random forest (Ho, 1995; Liaw, 2013; Hastie et al., 2008).

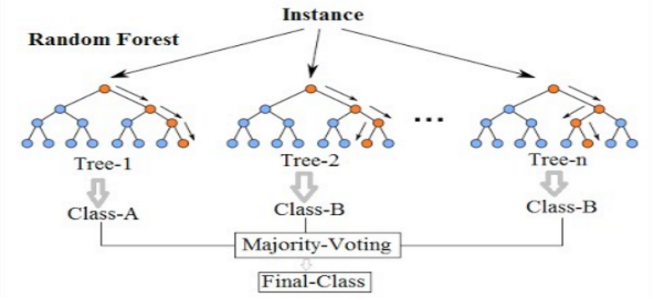


Figure 6 Random Forest Schema (Source: Liaw, 2013)

3.3.6 AdaBoost

AdaBoost learning algorithm is amongst ensemble ML methods whose implementation is based on boosting techniques. Boosting is developed as a method to provide higher learning performances by constructing a collection of "weaker" learners in a statistically purposeful way (Kégl, 2013; Friedman et al., 2000). AdaBoost can perform both regression and classification tasks.

AdaBoost's core method is based on training many so-called weak learners where a weak learner may be defined as a learner which is able to provide a prediction result that is incrementally more improved than a random/blind guess (Kégl, 2013; Zhang, 2004). The class of a weak learner used in the core of AdaBoost learning algorithm is a so-called decision stump. Decision stump may be best defined as a decision tree which consists of only one single node and two leaves. An individual tree is trained to pay specific attention to the weakness of only the previous tree. The weight of a sample misclassified by the previous tree will be boosted so that the subsequent tree focuses on correctly classifying the previously misclassified sample. The classification accuracy increases when more weak classifiers are added in series to the model (see Figure 7) (Chatterjee et al., 2019).

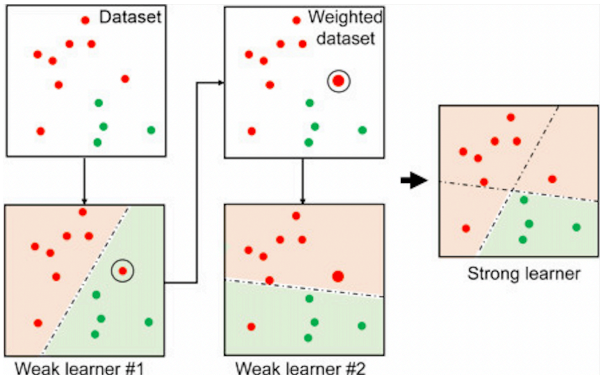


Figure 7 Implementation of AdaBoost on a Dataset (Source: Chatterjee et al., 2019)

3.4 Stacked Modeling

Leo Breiman, known for his work on classification and regression trees and random forests, formalized stacking in his 1996 paper on *Stacked Regressions* (Breiman et al., 1996). Although the idea originated by Wolpert (1992) under the name "Stacked Generalizations".

Stacking involves training a new learning algorithm to combine the predictions of several base learners. First, the base learners are trained using the available training data, then

a combiner or meta-algorithm, called the *super learner*, is trained to make a final prediction based on the predictions of the base learners. Such stacked model tends to outperform any of the individual base learners (Van Der Laan, 2007; Himmetoglu, 2017). See Figure 8.

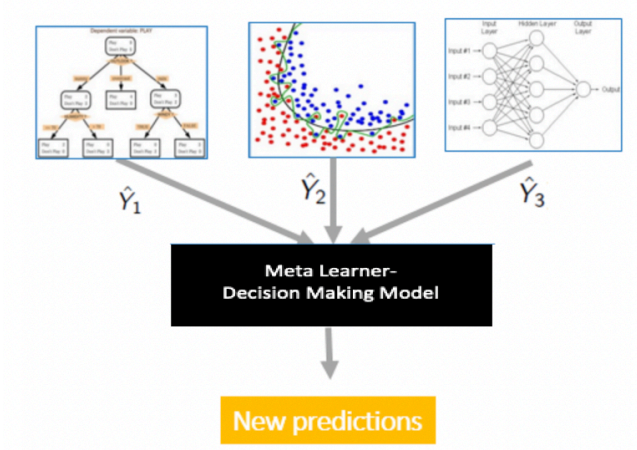


Figure 8 Concept Diagram of Stacked Modeling (Source: Van Der Laan, 2007)

3.5 Performance Evaluation of Regression ML Algorithms

The performance evaluation of ML algorithms is done by way of the so-called loss functions. Depending on the type of ML algorithm considered (i.e., classification or regression), loss functions are also categorized into either classification loss or regression loss (Grover, 2018). Figure 9 presents a summary of loss functions.

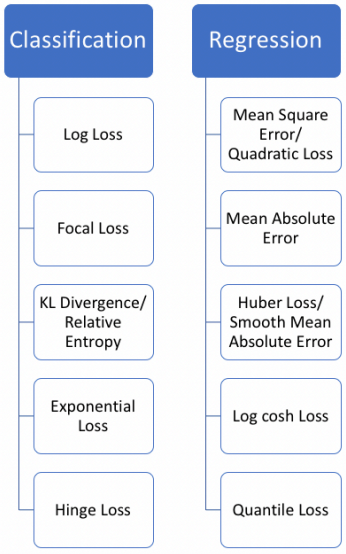


Figure 9 Summary of Regression and Classification Loss Functions (Source: Grover, 2018)

In this study, we applied regression loss functions including Mean Square Error (MSE), Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and R-squared (R^2).

Chapter 4. Data CETL

4.1 Data Collection

The seafood production is a multi-stage process which performs processing of various marine products, starting from their catching all the way to final canning and dispatching of the finished product (e.g., fish canning production as shown in Figure 10). At every stage of the process, certain measurements are to be taken and recorded, such as thawing and steam cooking duration and temperature or can count in the storage room.

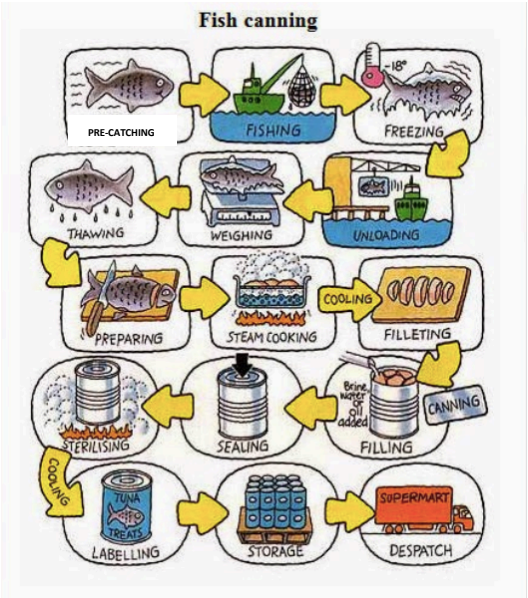


Figure 10 Scheme of Fish Canning Production (Source: Fish Canning, 2020)

Traditionally, this data has been collected manually and documented on paper media obviously being time-consuming and human error-prone, thus limiting or even precluding real-time analytics and item-level tracking capability. In order to overcome said shortages of "traditional informatics", the ThisFish Inc. has developed a unique software system "Tally" that enables seafood processors to automate and digitize all their production and quality control data (Figure 11 and Figure 12).

Figure 11 Moving from Traditional to Digitized Data Collection using "Tally" (Source: This Fish, 2019)

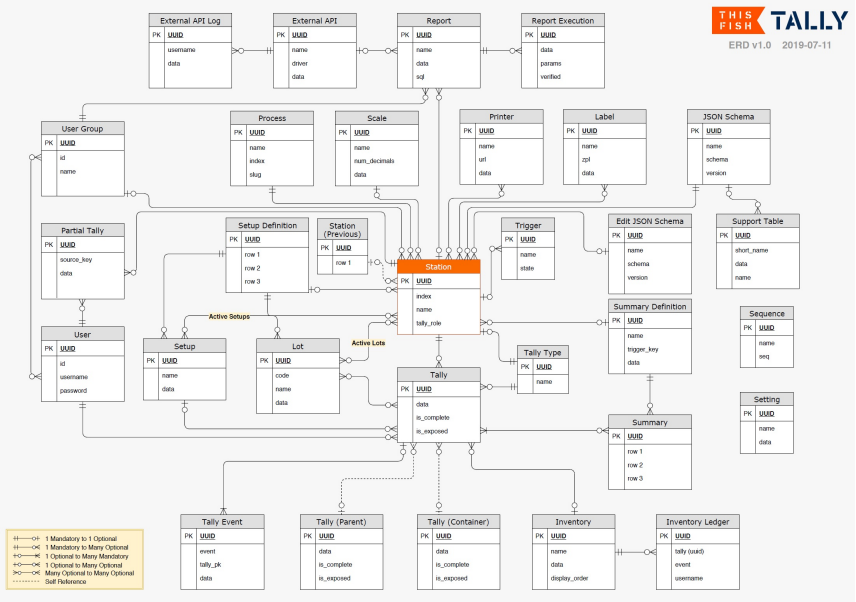


Figure 12 ERD Diagram of the TALLY Software System (Source: This Fish, 2019)

Figure 13 A Worker Using Tally Software System (Source: This Fish, 2019)

A large industrial tuna canning plant in Bangkok, Thailand, has now been utilizing the Tally software for almost three years. 30 workers are using the Tally on their hand-held devices (see Figure 13) to collect raw fish quality and process control data on everyday basis and have accumulated to-date a dataset of about 22 GB, spanning over a two-year period from January 2018 till November 2020.

4.2 Data Extraction

An initial source of data for this research was a damp file of the Tally PostgreSQL database which can be largely classified into two main groups, namely, process control parameters consisting of 167 parameters and raw material parameters consisting of 77 parameters, with metadata summarized in Table 53 of 0. The raw material data was atomic values while process control data was complex structures, with nested elements (see Table 54 of 1.1.1.1.1Appendix B). It required to design the two SQL queries which returned datasets of two different formats –

relational table for raw material parameters (see Box 1) and JSONB format for process control parameters.

Box 1 SQL Query for Raw Material Parameters

```sql
SELECT rm_code, can_code, fill_weight::FLOAT,
max(rm_ton_total)::FLOAT AS rm_ton_total,
max(rm_ton)::FLOAT AS rm_ton, sum(can_count)::INT AS can_count,
sum(meat)::FLOAT AS meat, max(rm_meat_total)::FLOAT AS rm_meat_total,
max(round((meat / rm_ton) * 100, 2))::FLOAT AS yield_fw,
count(*) FROM (SELECT data #>> '{lot,code,name}' AS rm_code,
data #>> '{pack,setup,can_code}' AS can_code,
round((data #>> '{pack,setup,fill_weight_g}')::NUMERIC, 2) AS fill_weight,
sum((data #>> '{seam,basket,can_count}')::INT) AS can_count
FROM stations_tally WHERE data #>> '{lot,pack_date}' = '2019-07-01'
AND station_id = '0fcdbe18-7b69-45d3-8537-9bff5bc756b3'
GROUP BY rm_code, can_code, fill_weight) tmp
JOIN LATERAL (SELECT round(fill_weight * can_count / 1000, 2) AS meat
LIMIT 1) tmp2 ON true JOIN LATERAL
(SELECT sum((data #>> '{dump,weight_kg}')::int) AS rm_ton_total
FROM stations_tally  WHERE data #>> '{lot,code,name}' = rm_code
AND data #>> '{lot,pack_date}' = '2019-07-01'
AND station_id = '5a6a24ea-de5e-410b-b189-82e0a1e4ec70') tmp3 ON true
JOIN (SELECT rm_code AS rm_code_2,
sum(round((tmp5.fill_weight::NUMERIC * tmp5.can_count / 1000), 2)) AS rm_meat_total
FROM (SELECT data #>> '{lot,code,name}' AS rm_code,
round((data #>> '{pack,setup,fill_weight_g}')::NUMERIC, 2) AS fill_weight,
sum((data #>> '{seam,basket,can_count}')::INT) AS can_count FROM stations_tally
WHERE data #>> '{lot,pack_date}' = '2019-07-01'
AND station_id = '0fcdbe18-7b69-45d3-8537-9bff5bc756b3'
GROUP BY rm_code, fill_weight) tmp5 GROUP BY rm_code) tmp4 ON tmp4.rm_code_2 = rm_code
JOIN LATERAL (SELECT round(((meat::NUMERIC / rm_meat_total) * rm_ton_total), 2)::NUMERIC AS rm_ton LIMIT 1)
tmp6 ON TRUE GROUP BY rm_code, can_code, fill_weight;
```

Box 2 SQL Query for Process Parameters

```sql
SELECT m.pk AS pk,
m.user as "user",
  (m.data#>>'{dump}')::jsonb AS dump,
  (m.data#>>'{rack}')::jsonb AS rack,
  (m.data#>>'{retort}')::jsonb AS retort,
  (m.data#>>'{pack}')::jsonb AS pack,
  (m.data#>>'{seam}')::jsonb AS seam,
  (m.data#>>'{lot}')::jsonb AS lot,
  (m.data#>>'{clean}')::jsonb AS clean,
  (m.data#>>'{precook}')::jsonb AS precook
FROM
(SELECT id AS pk,
(SELECT username FROM auth_user WHERE id = o.user_id) AS "user",
CASE WHEN (SELECT CASE WHEN y.retort_data IS NOT NULL THEN
jsonb_build_object('seam', x.seam_data, 'retort', y.retort_data)
ELSE jsonb_build_object('seam', x.seam_data) END FROM
(SELECT (data#>>'{seam}')::jsonb AS seam_data, container_id FROM
stations_tally  WHERE parent_id=o.container_id ORDER BY created DESC LIMIT 1) x LEFT JOIN lateral
(SELECT (data#>>'{retort}')::jsonb AS retort_data FROM stations_tally WHERE id=x.container_id)
y ON true) IS NOT NULL THEN data || (SELECT CASE WHEN y.retort_data IS NOT NULL THEN
jsonb_build_object('seam', x.seam_data, 'retort', y.retort_data) ELSE jsonb_build_object('seam', x.seam_data) END
FROM (SELECT (data#>>'{seam}')::jsonb AS seam_data, container_id FROM stations_tally
WHERE parent_id=o.container_id ORDER BY created DESC LIMIT 1) x LEFT JOIN lateral
(SELECT (data#>>'{retort}')::jsonb AS retort_data FROM stations_tally WHERE id=x.container_id) y
ON true) ELSE data END AS data FROM stations_tally AS o WHERE station_id='d084204c-8078-4958-a439-dcf771487516'
AND (data#>>'{lot,pack_date}')::DATE = '2019-07-01' ORDER BY created) AS m;
```

To serialize the objects in the Process Control Parameters dataset and to transform them further into relational tables, a complex script was applied to extract the attribute-value pairs (see Box 2).

4.3 Target Variable Selection

Given the objective of the study is maximizing the production yield, the later was chosen as the target variable. Provided that *Production Yield Value (PYV)* was only present in the Raw Material Parameters dataset but absent in the Process Control Parameters dataset, these two datasets have been joined to form a common Working dataset based on the *Can_Code* attribute in the Raw Materials dataset and *Can_ID_Code* attribute in the Process Control Parameters dataset which represent the same data elements.

4.4 Data Pre-Processing

In this sub-section, we present the main steps performed for data pre-processing. These include null values treatment, calculation of the *PYV* values based on several parameters in the Working dataset, followed by stratification of the Working dataset according to the six fish species. Finally, we discuss the mapping method applied to the categorical (qualitative) and string (range) parameters.

4.4.1 Yield Calculation

Specific circumstances of the primary data collection in the canning factory do not permit to directly relate the output product to the raw material used in its production due to the different granularity of these parameters. The problem of synchronizing the two said parameters in the Working dataset has been approached by a grouping exercise over *Dump_Weight*, *Cans_Total*, and *Pack_Weight* parameters as illustrated in Figure 14.

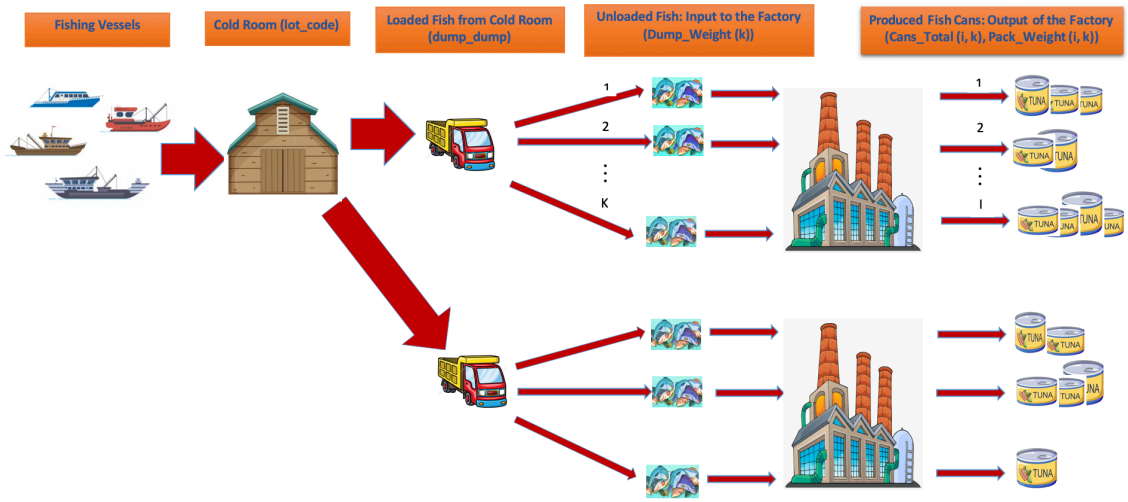


Figure 14 Steps of Computing Production Yield Value (*PYV*)

As shown in Figure 14, the correspondence is being established between the total produced fish meat as the product of total number of cans (*Cans_Total*) and their respective weight (*Pack_Weight*) and the raw fish meat dumped at the factory door (*Dump_Weight*) by every transferring truck from the cold room. This relationship is formalized in Equation (3).

$$production\ yield\ value\ (PYV) = \frac{Total\_Produced\_Meat}{Total\_Raw\_Material} = \frac{\sum_{k\in K}\sum_{i\in I}[Cans\_Total(i,k) * Pack\_Weight(i,k)]}{1000 * \sum_{k\in K} Dump\_Weight(k)}, \quad (3)$$

where $Dump\_Weight(k)$ is the k^{th} piece of raw fish meat from a given $dump\_dump$; $Cans\_Total(i,k)$ and $Pack\_Weight(i,k)$ are the number of cans and their pack weight, respectively, produced from $Dump\_Weight(k)$. Also, I is the total quantity of cans and K is the total amount of $Dump\_Weight(k)$ pieces. Corresponding Python code implementing the logic of Equation (3) is presented in Box 3.

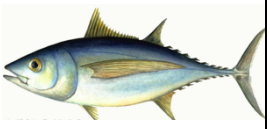
Box 3 Python Code for Computing *PYV*

```python
def calculate_yields_full(df, stations_dict, group_vars = ["lot_code_name", "dump_dump", "lot_size", "lot_pack_date"]):
    """
    Calculates the yields on a per rm-dump-lot_size-pack_date basis
    :return:
    """
    df_R_Full = pd.DataFrame()
    for rm_code_oi in tqdm.tqdm(df["lot_code_name"].unique(), desc="Calculating yields"):
        df_dumping = df.query(
            '(station_id == "{}") & (lot_code_name == "{}")'.format(stations_dict['dumping'], rm_code_oi))
        df_seaming = df.query(
            '(station_id == "{}") & (lot_code_name == "{}")'.format(stations_dict['seaming'], rm_code_oi))
        df_pack_weight = df.query('(lot_code_name == "{}")'.format(rm_code_oi))

        r = []
        group_vars_sub = group_vars[1:]
        for group_vals, dat in df_dumping.groupby(group_vars_sub):
            if len(group_vars_sub) == 1:
                group_vals = [group_vals]
            # Very occasionally there are duplicate values
            # Sum up the unique values to come up with the total raw material used in this group_vals
            raw_material = np.sum(list(set(dat["dump_weight_kg"].values.tolist())))
            # Build up the query str
            query_str = ""
            for i in range(len(group_vars_sub)):
                if len(query_str) > 0:
                    query_str = query_str + " & "
                query_str = query_str + '({} == "{}")'.format(group_vars_sub[i], group_vals[i])
            df_seaming_current = df_seaming.query(query_str)
            df_pack_weight_current = df_pack_weight.query(query_str)

            # This can be zero, which is indicates dropped data
            seaming_can_codes = [x for x in df_seaming_current["pack_setup_can_code"].unique() if not pd.isnull(x)]
            packing_can_codes = [x for x in df_pack_weight_current["pack_setup_can_code"].unique() if not pd.isnull(x)]
            if (len(seaming_can_codes) == 0) | (len(packing_can_codes) == 0):
                continue

            # Sometimes they don't overlap their cans, so need to intersect
            can_codes_use = set(seaming_can_codes).intersection(packing_can_codes)

            # Calculate the total meat for all can_codes used in the set of group_vals in this iteration
            total_meat = 0
            for can_code in can_codes_use:
            # Calculate the meat, per can_code, at every iteration and updates the total_meat at the end of it by
            # summing over all these used can_codes
                cans_total = df_seaming_current.query('pack_setup_can_code == "{}"'.format(can_code))["seam_basket_can_count"].sum()
                # NaN happens when pack_weight is NaN
                pack_weight = df_pack_weight_current.query('pack_setup_can_code == "{}" '.format(can_code))["pack_setup_fill_weight_g"].mean()
                total_meat += pack_weight * cans_total
            yield_val = total_meat / 1000 / raw_material
            r.append([rm_code_oi] + list(group_vals) + [raw_material, cans_total, pack_weight, yield_val])
        colnames_use = list(group_vars) + ["raw_material","cans_total","pack_weight", "yield_val"]
        df_R = pd.DataFrame(r, columns=colnames_use)
        df_R_Full = df_R_Full.append(df_R)

    return df_R_Full
```

4.4.2 Data Stratification

There are many fish species used in the seafood production industry. In this thesis, however, six major tuna fish species have been studied consisting of Albacore (AL), Yellow Fin (YF), Skipjack (SK), Big Eye (BE), Longtail (TG), and Bonito (BT). The description of these six species is given in Table 2.

Table 2 Fish Species Characteristics

Fish species	Image	Max Length	Max Weight	Habitat	Acronym
SKIPJACK (*Katsuwonus pelamis*)		108 cm/3.5 feet	33 kg/73 lbs	tropical areas of the Atlantic, Indian and Pacific Oceans, with the greatest abundance seen near the equator	SK
ALBACORE (*Thunnus alalunga*)		130 cm/4.3 feet	40 kg/85 lbs	temperate and tropical waters of the Atlantic, Pacific, and Indian Oceans, as well as the Mediterranean Sea	AL
YELLOWFIN (*Thunnus albacares*)		205 cm/6.7 feet	194 kg/427 lbs	deep offshore waters throughout the Pacific, Atlantic and Indian Oceans	YF
BIGEYE (*Thunnus obesus*)		230 cm/7.5 feet	210 kg/462 lbs	all tropical and temperate oceans (but not in the Mediterranean), the Indian and Pacific Oceans	BE
LONGTAIL (*Thunnus tonggol*)		145cm/4.75 feet	36 kg/79 lbs	tropical Indo-West Pacific waters	TG
BONITO (*Sarda sarda*)		75 cm/2.5 feet	6 kg /12 lbs	common in shallow waters of the Atlantic Ocean, the Mediterranean Sea and the Black Sea	BT

Owing to the hypothesis that the predictive models for target variable, *PYV*, may vary for different fish species, the Working dataset was stratified based on the six fish species considered in this study as discussed in the above table. Accordingly, further data analysis and ML modeling

was performed on both the entire Working dataset and stratified Working datasets for each individual fish species.

4.4.3 Data Mapping

One of the most important features which should be included in the analysis is the "*lot_size*" attribute that shows the size of the fish caught by the fishing vessel and in the frozen raw meat lots used in the canning production. However, this attribute has been recorded in the original Raw Material Parameters dataset and migrated from there into the Working dataset in the form of strings (ranges), not atomic values, e.g., '3.0-4.0' which denotes fish sizes between 3.0 and 4.0. It is worth to mention that these ranges do not have units and are defined based on some fishery standards. Moreover, the *lot_size* ranges overlap, e.g., '1.0-1.4', '1.3-1.8', '1.4-1.8', '1.8-2.3'. As the third data pre-processing step, every *lot_size* range was mapped into the mean value of its upper- and lower-bound. Table 3 shows the resulting dictionary for each fish species in the study.

Table 3 Dictionary for *lot_size* Mapping

Fish Species	*lot_size* Dictionary
AL	{'4.0-6.0': **5.0**, '10.0-12.0': **11.0**, '9.0-10.0': **9.5**, '15.0-20.0': **17.5**, '12.0-15.0': **13.5**, '7.0-9.0': **8.0**, '2.0-3.0': **2.5**, '4.3-5.0': **4.65**, '6.0-7.0': **6.5**, '3.0-4.0': **3.5**, '6.0-7.0': **6.5**, '20.0-30.0': **25.0**, '10.0-20.0': **15.0**, '0.8-1.0': **0.9**, '<2.0': **1.0**}
SK	{'5.0-6.0': **5.5**, '3.4-4.2': **3.8**, '4.3-5.0': **4.65**, '6.0-7.0': **6.5**, '2.9-3.4': **3.15**, '1.0-1.4': **1.2**, '1.8-2.4': **2.1**, '2.4-2.9': **2.65**, '7.0-9.0': **8.0**, '1.4-1.8': **1.6**, '5.0-6.0': **5.5**, '0.8-1.0': **0.9**, '<0.8': **0.4**, '6.0-7.0': **6.5**, '<0.3': **0.15**, '7.0-9.0': **8.0**, '0.7-1.3': **1.0**, '9.0-12.0': **10.5**, '1.3-1.8': **1.55**, '1.8-3.4': **2.6**, '3.4-5.0': **4.2**, '5.0-9.0': **7.0**, '<2.0': **1.0**}
BE	{'5.0-6.0': **5.5**, '9.0-12.0': **10.5**, '3.4-4.2': **3.8**, '1.8-2.4': **2.1**, '1.4-1.8': **1.6**, '5.0-9.0': **7.0**, '1.8-3.4': **2.6**, '3.4-5.0': **4.2**, '15.0-20.0': **17.5**, '9.0-15.0': **12.0**, '9.0-10.0': **9.5**, '10.0-20.0': **15.0**, '4.0-6.0': **5.0**}
BT	{'3.4-4.2': **3.8**, '1.8-2.4': **2.1**, '6.0-7.0': **6.5**, '5.0-6.0': **5.5**, '1.3-1.8': **1.55**, '0.7-1.3': **1.0**, '1.4-1.8': **1.6**, '1.8-3.4': **2.6**, '3.4-5.0': **4.2**, '1.0-1.4': **1.2**, '5.0-6.0': **5.5**, '0.8-1.0': **0.9**, '0.5-0.7': **0.6**}
YF	{'3.4-5.0': **4.2**, '1.4-1.8': **1.6**, '5.0-9.0': **7.0**, '1.8-3.4': **2.6**, '15.0-20.0': **17.5**, '1.0-1.4': **1.2**, '1.8-2.4': **2.1**, '9.0-15.0': **12.0**, '0.8-1.0': **0.9**, '<0.3': **0.15**, '9.0-12.0': **10.5**, '7.0-9.0': **8.0**, '3.4-4.2': **3.8**, '7.0-9.0': **8.0**, '5.0-6.0': **5.5**, '20.0-30.0': **25.0**, '9.0-10.0': **9.5**, '30 UP': **35.0**, '10.0-20.0': **15.0**}
TG	{'3.4-4.2': **3.8**, '1.8-2.4': **2.1**, '5.0-6.0': **5.5**, '7.0-9.0': **8.0**, '4.3-5.0': **4.65**, '1.0-1.4': **1.2**, '1.4-1.8': **1.6**, '0.7-1.3': **1.0**, '0.8-1.0': **0.9**, '1.3-1.8': **1.55**, '1.8-3.4': **2.6**, '3.4-5.0': **4.2**, '5.0-6.0': **5.5**, '3.0-4.0': **3.5**}

Other data mapping operations have been performed for redundancy elimination in categorical features as explained in Subsection 4.5.3.

4.5 Data Cleansing

After the pre-processing step, we need to detect and correct or remove inaccurate/corrupted records from the Working dataset. To this aim, we analyzed the dataset in three ways: 1) to filter the outliers in the numeric features 2) to deal with the null values; and 3) to eliminate redundancy of duplicated values in categorical features. The details of these procedures are discussed in this sub-section.

4.5.1 Outlier Removal

Two types of numeric data have been noted: the features with temporal characteristics (e.g., cooking time, steam time, *etc*.) and non-temporal features, such as rack weight, precook cooker temperature, *etc*. Definition of these parameters have been provided in Table 53 of 0. Regarding the first data type and from practical considerations, it was assumed that the duration of certain technological operations (e.g., thawing, steaming, cooking, *etc.*) cannot exceed 12 hours, and instances with longer duration were considered as outlier.

For the second data type, the empirical 3-sigma rule (Gagnon et al., 2009; Aivodji et al., 2019) has been applied. Outliers of both types had been filtered from the Working dataset. The details of the filtering techniques are summarized in Table 4.

Table 4 Outlier Removal Criteria

Feature	Criterion for Outlier Filtering
precook_steam_time_min precook_spray_time_min precook_cooking_time_min	Time duration more than 12 hours
clean_bb_after_chill_temps_c_avg rack_internal_temps_c_avg precook_cooker_temp_c precook_after_spray_temps_c_avg precook_bb_temps_c_avg rack_weights_kg_avg rack_total_pans rack_fish_per_pan rack_pans_per_rack_avg	Outside 3-standard deviation range

Additional filtration on data will take place as the result of normality test in Section 5.2.2 of Chapter 5.

4.5.2 Null Values Treatment

The next step of data cleansing entails verifying the presence/absence of feature values in the Working dataset. Several columns in the dataset contained Null values denoting missing observations. As a solution to this problem, columns with more than 90% of Null values were completely discarded while for others only rows with Null values were deleted.

4.5.3 Redundancy Elimination

Categorical features appeared in several different spelling ways in the original Raw Material Parameters dataset and, as such, migrated into the Working dataset (e.g., 'POLE&LINE' and 'POLE & LINES'). The dictionary for redundancy elimination in three categorical features

(i.e., catching method (*CT_Method*), catching area (*CT_Area*) and fishing vessel's flag (*FV_Flag*)) is presented in Table 5, Table 7, and Table 9.

A summary of the unique levels of categorical features for each of the six fish species is given in Table 6, Table 8, and Table 10.

Table 5 Dictionary for Redundancy Elimination of *CT_Method*

Original Value	Replacement Value
Empty string	Removed
'POLE&LINE', 'POLE & LINES', 'POLE&LINE ', 'POLE & LINES'	'POLE & LINE'
'LONGLINE'	'LONG LINE'
'TROLL&LINE', 'TROLLED', 'TROLLING', ' TROLL&LINE ', 'TROLL LINE'	'TROLL & LINE'
'PURSE SEINING', 'PURSE SEINE '	'PURSE SEINE'

Table 6 Unique Values of *CT_Method* Upon Redundacy Elimination

Fish Species	Unique Values
AL	'POLE & LINE', 'TROLL & LINE', 'LONG LINE', 'TROLL-JIG/POLE&LINE'
SK	'PURSE SEINE', 'POLE & LINE', 'HAND LINE', 'FAD FREE PURSE SEINE'
BE	'PURSE SEINE', 'POLE & LINE'
BT	'PURSE SEINE', 'POLE & LINE'
YF	'POLE & LINE', 'HAND LINE', 'PURSE SEINE'
TG	'PURSE SEINE'

Table 7 Dictionary for Redundancy Elimination of *CT_Area*

Original Value	Replacement Value
Empty string	Removed
'WESTERN PACIFIC OCEAN', 'WESTERN PACIFIC '	'WESTERN PACIFIC'
'INDIAN OCEAN ', ' INDIAN OCEAN'	'INDIAN OCEAN'
'NORTHWEST PACIFIC OCEAN'	'NORTHWEST PACIFIC'
'EASTERN PACIFIC OCEAN'	'EASTERN PACIFIC'

Table 8 Unique Values of *CT_Area* Upon Redundacy Elimination

Fish Species	Unique Values
AL	'SOUTHWEST PACIFIC' 'NORTHWEST PACIFIC' 'WESTERN PACIFIC' 'NORTHEAST PACIFIC' 'EASTERN PACIFIC'
SK	'WESTERN PACIFIC' 'INDIAN OCEAN' 'NORTHWEST PACIFIC' 'EASTERN PACIFIC'
BE	'WESTERN PACIFIC' 'INDIAN OCEAN'
BT	'WESTERN PACIFIC' 'JAVA SEA' 'INDIAN OCEAN'
YF	'INDIAN OCEAN' 'WESTERN PACIFIC'
TG	'WESTERN PACIFIC' 'JAVA SEA'

Table 9 Dictionary for Redundancy Elimination of *FV_Flag*

Original Value	Replacement Value
Empty string	Removed
'MARSHAIIL ', 'MARSHALL ISLANDS'	'MARSHALL IS'
'REPUBLIC OF KOREA ‘, 'REPUBLIC OF KOREA'	'KOREA'
'INDIAN OCEAN ', ' INDIAN OCEAN'	'INDIAN OCEAN'
' MALDIVES', 'MADIVES',	'MALDIVES'
'WESTERN PACIFIC OCEAN'	'WESTERN PACIFIC'
'PANAMA '	'PANAMA'
'FSM'	'MICRONESIA'
' NEW ZEALAND'	'NEW ZEALAND'

Table 10 Unique Values of *FV_Flag* Upon Redundacy Elimination

Fish Species	Unique Values
AL	'NEW ZEALAND' 'USA' 'JAPAN' 'FIJI' 'TAIWAN' 'CHINA'
SK	'MALDIVES' 'TAIWAN' 'JAPAN' 'PNG' 'USA' 'MARSHALL ISLANDS' 'MICRONESIA' 'INDONESIA' 'KOREA' 'NAURU' 'KIRIBATI'
BE	'MICRONESIA' 'KOREA' 'MALDIVES' 'CHINA' 'MARSHALL IS' 'TAIWAN' 'NAURU' 'USA'
BT	'INDONESIA' 'MALDIVES' 'KOREA'
YF	'INDONESIA' 'MALDIVES' ' MALDIVES' 'CHINA' 'MICRONESIA' 'USA' 'MARSHALL IS' 'KOREA' 'SOLOMON ISLANDS'
TG	'INDONESIA'

As a result of the above steps, a fully operational Working dataset was formed ready for the next phases of the framework, namely, data analysis, feature extraction, and modeling steps.

Chapter 5. Exploratory Data Analysis (EDA)

Once the Working dataset has been pre-processed into a usable format, the next phase of the data science framework is Exploratory Data Analysis (EDA). In this chapter and in order to perform this phase, we assessed the data, applied various statistical analysis techniques on the Working dataset, visualized the results in different ways to gain a deeper understanding and insights into the data required for predictive modeling in the next phases of this research.

The process of EDA is applied to the entire Working dataset as well as to the stratified datasets according to the six fish species (See Table 2). We first explore a hypothesis of the normal distribution of the target variable *PYV*. This is important to select appropriate statistical tests and techniques (e.g., parametric vs. non-parametric tests) in the analysis of different features and their effect on the target variable.

5.1 Normality Analysis of the *PYV* for All Fish Species

To test the normality of *PYV*, we need to compare the theoretical normal distribution with the actual distribution of this variable in the dataset. Broadly, two main methods are used for this purpose: a) graphical and b) statistical. Common graphical methods are Q-Q Probability Plot and Histogram Visualization. Statistical methods include different tests (e.g., W/S test, Jarque-Bera test, Shapiro-Wilks test, Kolmogorov-Smirnov test, D'Agostino test). In this section, the results of Q-Q plotting and Histogram visualization as graphical methods and statistical Kolmogorov-Smirnov (KS) test (as the most suitable for the size of the datasets) are presented.

5.1.1 Q-Q Plot of *PYV* Distribution

The distribution pattern of *PYV* using its Q-Q plot is shown in Figure 15. As it is seen in Figure 15, the majority of the data instances fall on, or reasonably close to, the identity line (y=x) and, therefore, *PYV* for all fish species fairly obeys the normal distribution.

5.1.2 Histogram of *PYV*

We have as well derived histogram of *PYV* distribution for all fish species in order to visually verify the normality. The result is shown in Figure 16.

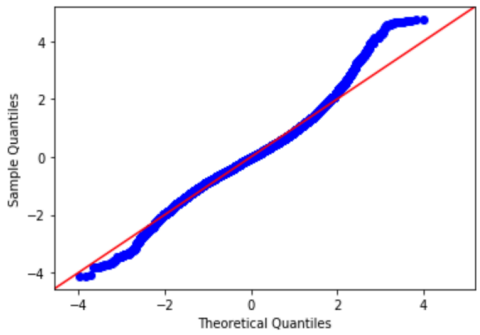


Figure 15 Q-Q Plot of *PYV* for All Fish Species

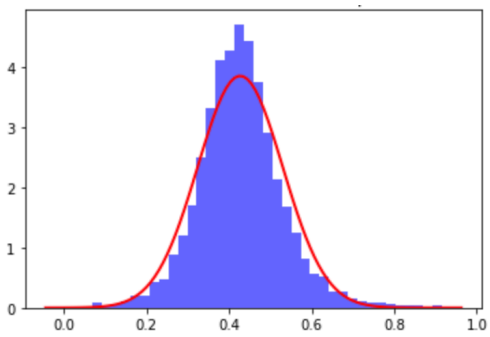


Figure 16 Histogram of *PYV* for All Fish Species

It can be seen that *PYV* of all fish species well aligns with the normal distribution pattern.

5.1.3 Kolmogorov-Smirnov (KS) Test of *PYV*

We applied the KS test on the dataset with all fish species to evaluate the null hypothesis:

H_0 = the *PYV* follow the normal distribution.

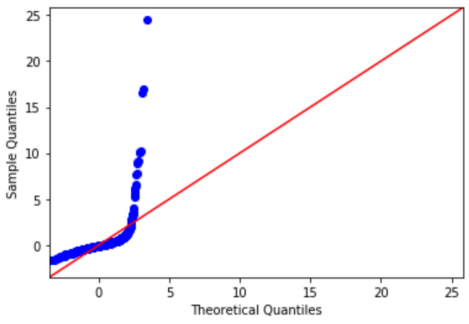
As a result of this test, *p*-value was 0.05879, which is greater than the threshold value of 0.05. Accordingly, we failed to reject the null hypothesis of the normal distribution.

5.2 Normality Analysis of *PYV* for Stratified Fish Species

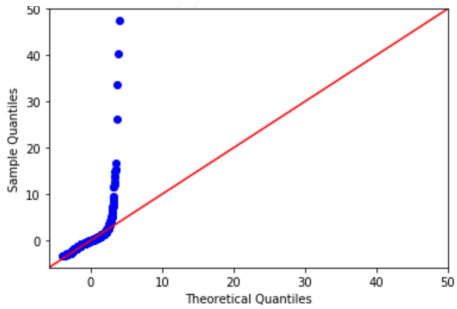
The same three techniques have been applied to test the normality of *PYV* for stratified Working datasets.

5.2.1 Q-Q Plot of Stratified *PYV* Distribution

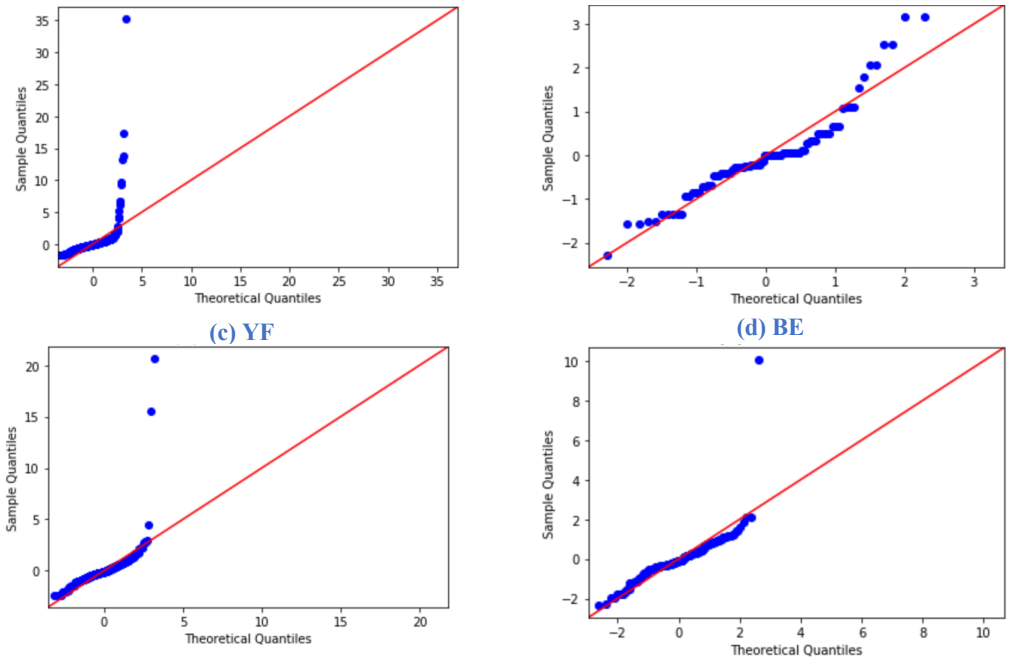
Figure 17 shows the Q-Q Plots of *PYV* for each fish species in the study.



(a) AL



(b) SK



(c) YF (d) BE

(e) BT (f) TG

Figure 17 Q-Q Plots of *PYV* for Each Fish Species

It can be noted that *PYV* for stratified datasets deviate from the normal distribution.

5.2.2 Kolmogorov-Smirnov (KS) Test of *PYV*

Since *PYV* for all fish species follow the normal distribution and parametric tests are generally more powerful than non-parametric tests, we attempted to normalize the *PYV* of stratified datasets while keeping the majority of data instances. In order to achieve this goal, we applied an iterative procedure to check different sigma-away (SA) values, starting from SA=3.00, with constant decrease of 0.01 for filtering *PYV*, until KS test would fail to reject the null hypothesis of normal distribution for stratified PYV. The pseudo-code for performing this procedure is shown in Box 4.

Box 4 Python Code for Filtration of *PYV*

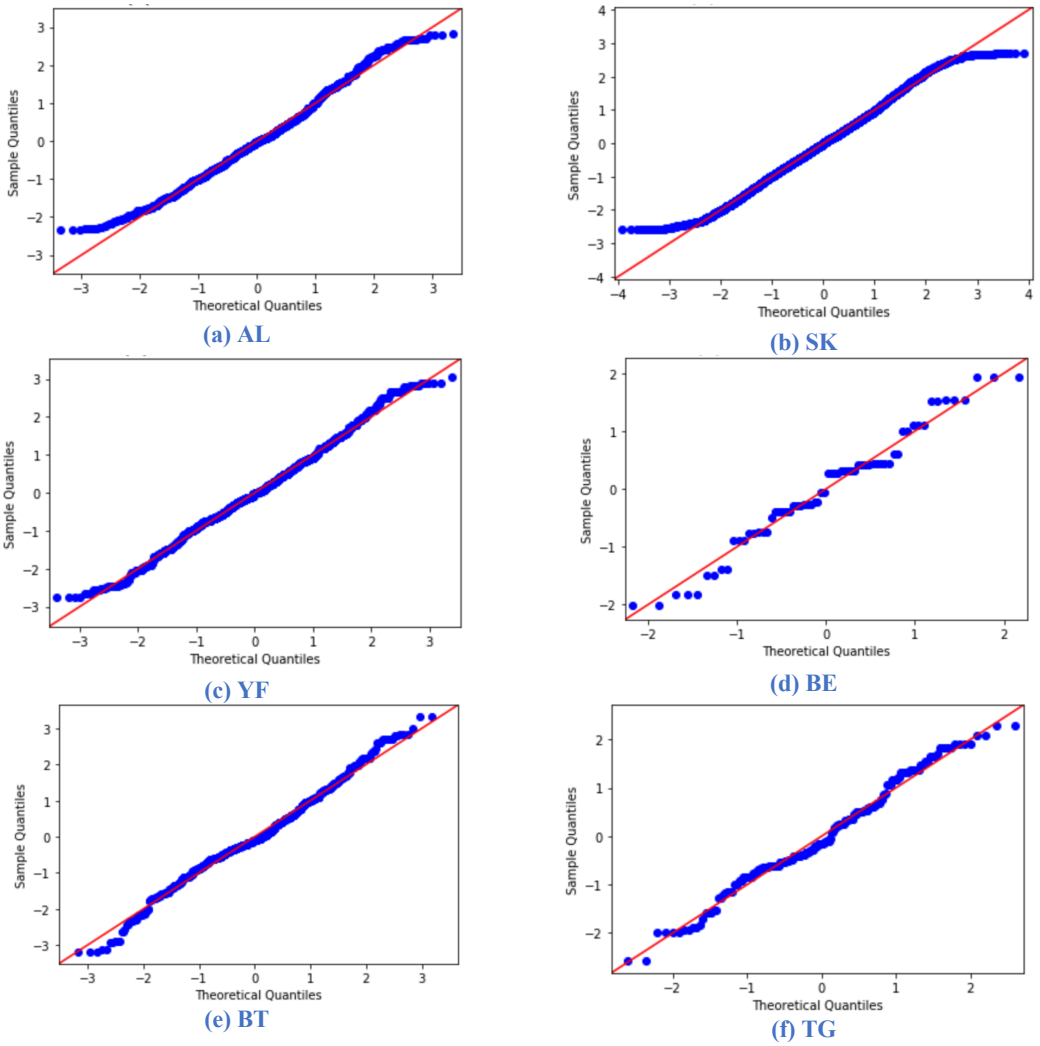
```
def calculate_pyv_filtration(df, sa_start = 3, sa_step = 0.01):
    """
    Calculates the filtration for given df based on shrinking sigma-away window from the
    mean value to obtain filtered df_filt which obeys normal distribution.
    Algorithm starts with sa_start sigma-away filtration and continues with step size
    given by sa_step.
    :return:
    """
    df['pyv_zscore'] = stats.zscore(df['pyv'])
    pvalue_ks = (stats.kstest((df['pyv']-df['pyv'].mean())/df['pyv'].std(),'norm').pvalue)
    sa = sa_start
    while pvalue_ks < 0.05:
        df = df[(df['pyv_zscore']>=-sa) & (df['pyv_zscore']<=sa)]
        pvalue_ks = (stats.kstest((df['pyv']-df['pyv'].mean())/df['pyv'].std(),'norm').pvalue)
        sa = sa - sa_step

    return df, sa, pvalue_ks
```

Final values of sigma for filtering *PYV* per fish species and also the corresponding *p*-values of KS test are presented in Table 11.

Table 11 Final Sigma-away Values for PYV and p-values of KS Test

Fish Species	Sigma-away Value	*p*-value
AL	0.835	0.0593
SK	1.671	0.0668
YF	1.182	0.0563
BE	0.974	0.0525
BT	2.170	0.0579
TG	1.560	0.0518

Corresponding Q-Q Plots of *PYV* after this filtration and given the final sigma-away values from Table 1 are shown in Figure 18.



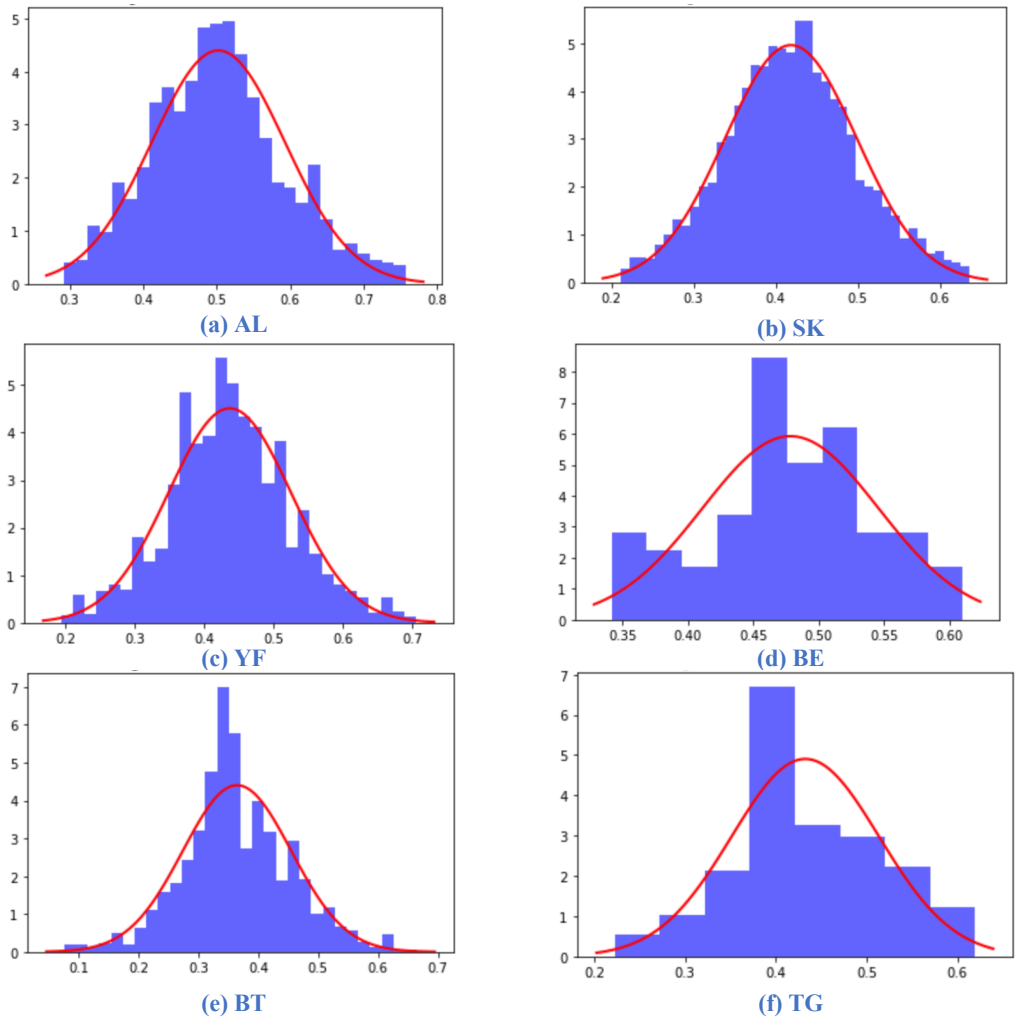
(a) AL (b) SK (c) YF (d) BE (e) BT (f) TG

Figure 18 Q-Q Plots of *PYV* for Each Fish Species after Filtration

As observed in Figure 18, the distribution for each fish species fairly closely follows the normal distribution.

5.2.3 Histogram of *PYV*

The Histogram visualization of *PYV* for each fish species after filtration is presented in Figure 19.



(a) AL (b) SK (c) YF (d) BE (e) BT (f) TG

Figure 19 Histogram of *PYV* for Each Fish Species after Filtration

As it is seen in Figure 5, three fish species (namely, SK, AL, and YF) better approximate the theoretical bell curve which can be attributed to the fact that these species have the highest number of data instances in the Working dataset (see Table 12).

Table 12 Number of Data Instances Before and After *PYV* Filtration for Each Fish Species

Fish Species	Before Filtration	After Filtration
AL	2720	2495
SK	23633	22554
YF	2922	2893
BE	88	66
BT	1333	1313
TG	231	216

5.3 Data Analysis of Process Control Parameters

Next step of the EDA is to analyze the relationship between *PYV* and process control parameters. To do so, Lasso regression coefficients and Pearson's correlation coefficient test as statistical methods and Heatmap visualization technique have been applied. Further definition of these process control parameters have been provided in Table 53 of 0.

5.3.1 Lasso Regression Coefficients

Lasso regression model to identify process control parameters with the higher impact on *PYV* has been used. The scaled Lasso regression coefficients are summarized in Figure 20.

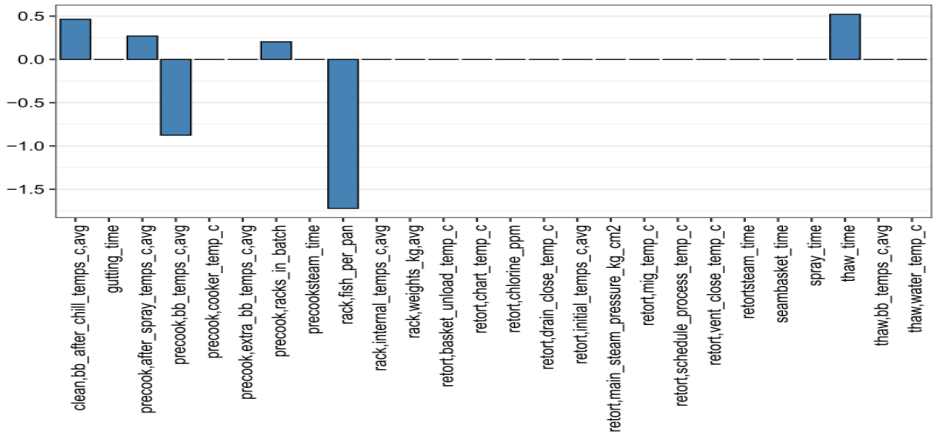


Figure 20 Lasso Regression Coefficients for Process Control Parameters and *PYV*

As follows from Figure 20, the following 6 process controls parameters appear with non-zero Lasso regression coefficients meaning that they are associated with the higher *PYV* for all fish species: *Clean_bb_after_chill_temps_C_avg*, *Precook_after_spray_temps_C_avg*, *Precook_bb_temps_C_avg*, *Precook_racks_in_batch, Rack_fish_per_pan, Thaw_time_min*.

5.3.2 Pearson Correlation Coefficient Values

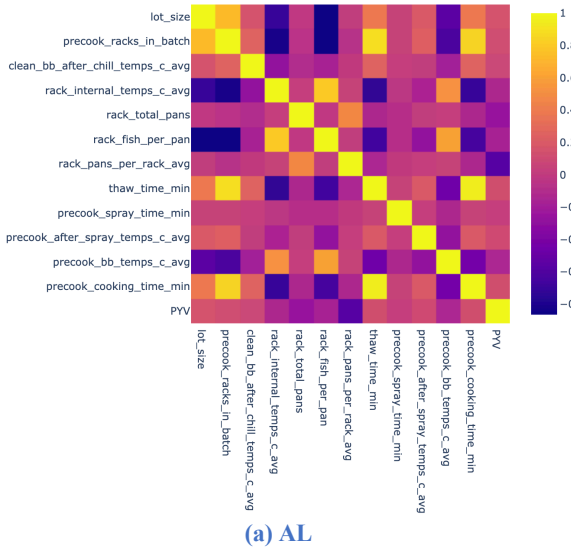
Pearson's test was applied to the set of process control parameters with the highest Lasso regression coefficients and some derived parameters of interest (e.g., *Precook_spray_time_min* and *Precook_cooking_time_min* which are time-durations computed based on their recorded initial and end time-stamps). The results are shown in Table 13.

Table 13 Pearson's Correlation Coefficients of *PYV* and Process Control Parameters for Each Fish Species

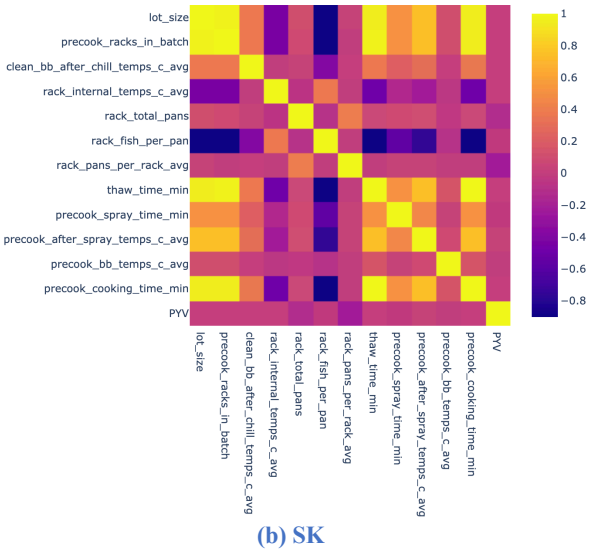
Process Control Parameter	Correlation Value for Each Fish Species with *PYV*					
	AL	SK	YF	BE	BT	TG
Lot_size	+0.148	+0.012	+0.061	+0.131	+0.090	-0.144
Precook_racks_in_batch	+0.121	+0.013	+0.074	-0.108	+0.056	-0.133
Clean_bb_after_chill_temps_c_avg	+0.0737	+0.005	+0.041	+0.021	+0.026	+0.008
Rack_internal_temps_c_avg	-0.123	+0.002	-0.055	+0.051	-0.044	+0.174
Rack_total_pans	-0.249	-0.117	-0.114	-0.025	-0.085	-0.045
Rack_fish_per_pan	-0.165	-0.016	-0.089	+0.032	-0.050	+0.103
Rack_pans_per_rack_avg	-0.568	-0.236	-0.337	-0.192	-0.286	-0.092
Thaw_time_min	+0.124	+0.008	+0.105	-0.177	+0.062	-0.126
Precook_spray_time_min	+0.032	-0.014	+0.019	+0.115	+0.049	-0.112
Precook_after_spray_temps_c_avg	+0.100	+0.020	+0.091	-0.296	+0.089	-0.071
Precook_bb_temps_c_avg	-0.141	-0.008	-0.037	-0.090	+0.043	-0.126
Precook_cooking_time_min	+0.123	+0.006	+0.107	-0.177	+0.063	-0.122

5.3.3 Heatmap of Pearson's Correlation Coefficients

Heatmap technique for visualization of Pearson's correlation coefficients of *PYV* and process control parameters which have been listed in Table 13 and also Pearson's correlation coefficients among process control parameters are shown in Figure 21.



(a) AL



(b) SK

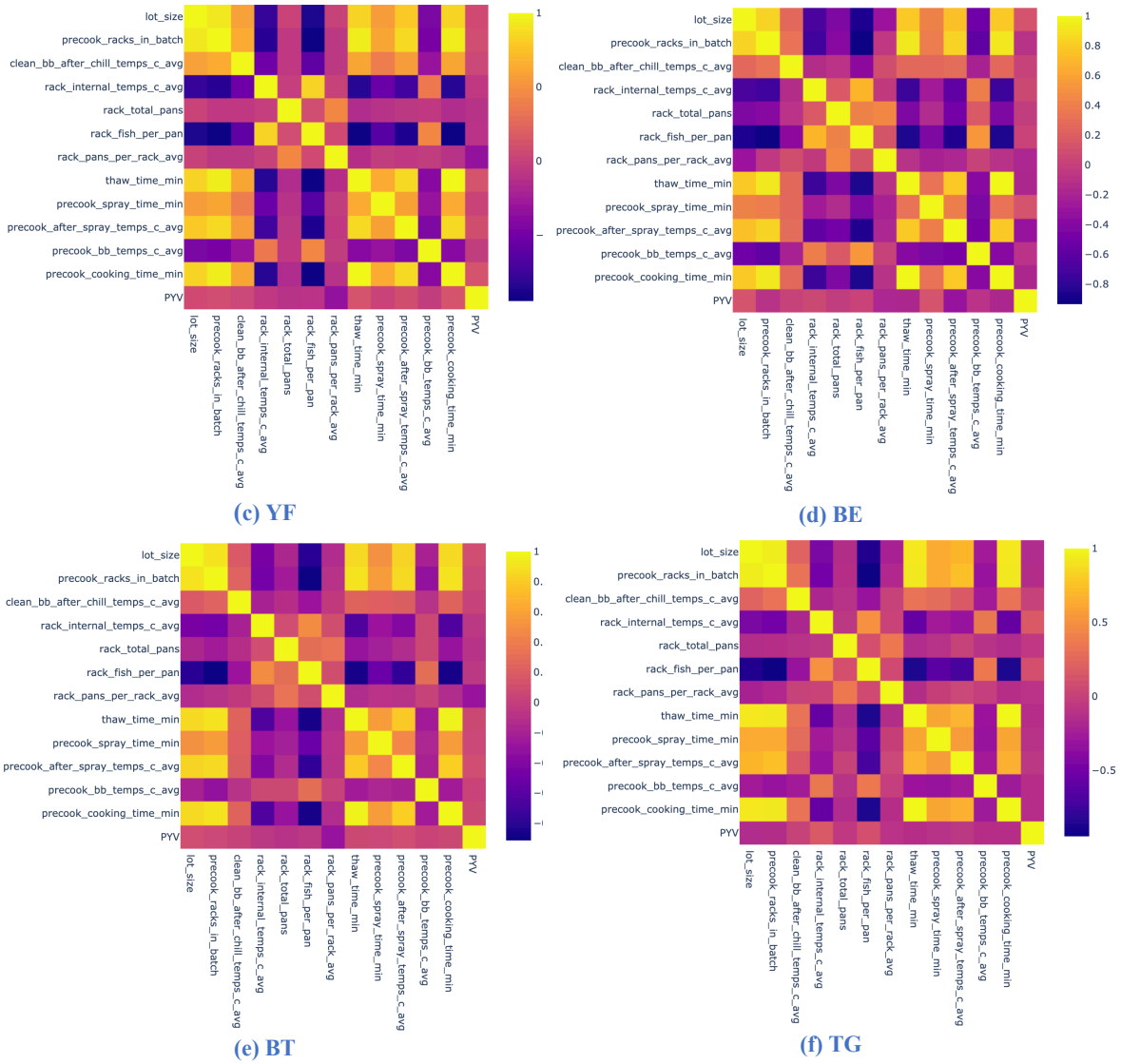


Figure 21 Heatmap of Pearson's Correlation Coefficients of Process Control Parameters and *PYV* for Each Fish Species

According to Table 13 and Figure 21, we can identify the impact of each process control parameter on *PYV* for each fish species. For instance, *lot_size* and *thaw_time_min,* have high positive correlation with *PYV*, whereas *rack_total_pans* and *rack_fish_per_pan* have high negative correlation with *PYV*. Also, *Clean_bb_after_chill_temps_c_avg* and *Precook_bb_temps_c_avg* are low correlated with *PYV*. These findings should be carefully considered in setting the values of process control parameters.

5.4 Data Analysis of Raw Material Parameters for All Fish Species

In this section, we analyze relationship between different levels of categorical Raw Material parameters (i.e., *CT_Method*, *CT_Area* and *FV_Flag*) and *PYV* and *lot_size* for all fish species. Provided the results of the previous sections, we applied ANOVA test, Lasso regression coefficient method and summary statistics such as Mean, Median, and Standard Deviation.

5.4.1 *CT_Method*

CT_Method is the parameter that characterizes a particular method used for catching fish on the ocean with the levels like 'HAND LINE' or 'POLE AND LINE'.

5.4.1.1 ANOVA Test

We applied ANOVA test with the following null hypothesis:

H_0 = *PYV* for different levels of *CT_Method* are equal.

As a result of this test, *p*-value was zero, which is less than the threshold value of 0.05. Accordingly, we rejected the above null hypothesis, i.e., *PYV* is different for different levels of *CT_Method* for all fish species.

5.4.1.2 Summary Statistics of *PYV* for All Fish Species

To study the variation of *PYV* across different levels of *CT_Method* for all fish species, we have computed summary statistics of *PYV*. The findings are presented in Table 14 and visualized in Figure 22.

Table 14 Summary Statistics of *PYV* for Levels of *CT_Method* for All Fish Species

CT_Method	Mean Value	Median Value	STD
'LONG LINE'	0.5810	0.5852	0.0828
'TROLL & LINE'	0.4793	0.4887	0.0869
'HAND LINE'	0.4485	0.4377	0.0546
'POLE & LINE'	0.4282	0.4266	0.0880
'PURSE SEINE'	0.4160	0.4158	0.0803

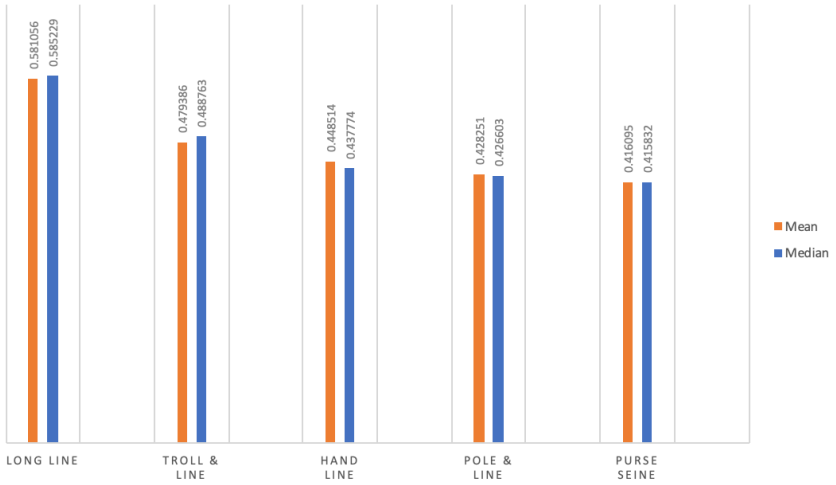


Figure 22 Summary Statistics of *PYV* across Levels of CT_*Method* for All Fish Species

It can be concluded that the highest mean value of *PYV* is associated with 'LONG LINE' catching method while 'PURSE SEIN' catching method is associated with the lowest mean value of *PYV*.

5.4.1.3 Summary Statistics of *lot_size* for All Fish Species

To study the variation of *lot_size* across levels of *CT_Metho*d, we have calculated summary statistics of *lot_size* for all fish species. Table 15 summarizes and Figure 23 visualizes these results.

Table 15 Summary Statistics of *lot_size* for Levels of *CT_Method* for All Fish Species

CT_Method	Mean value	Median value	STD
'LONG LINE'	19.0391	17.5	3.4078
'TROLL & LINE'	5.0406	5.0	1.8559
'HAND LINE'	8.3312	1.2	8.3512
'POLE & LINE'	4.7791	3.8	4.1084
'PURSE SEINE'	3.5995	3.8	1.7349

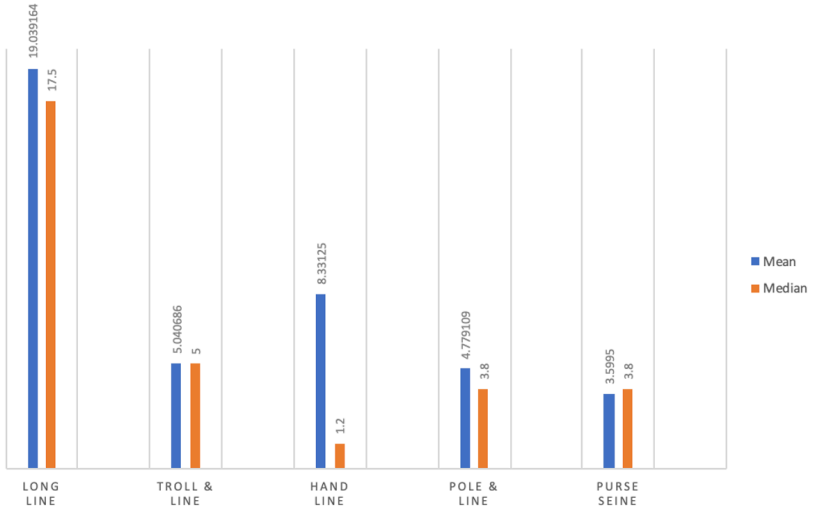
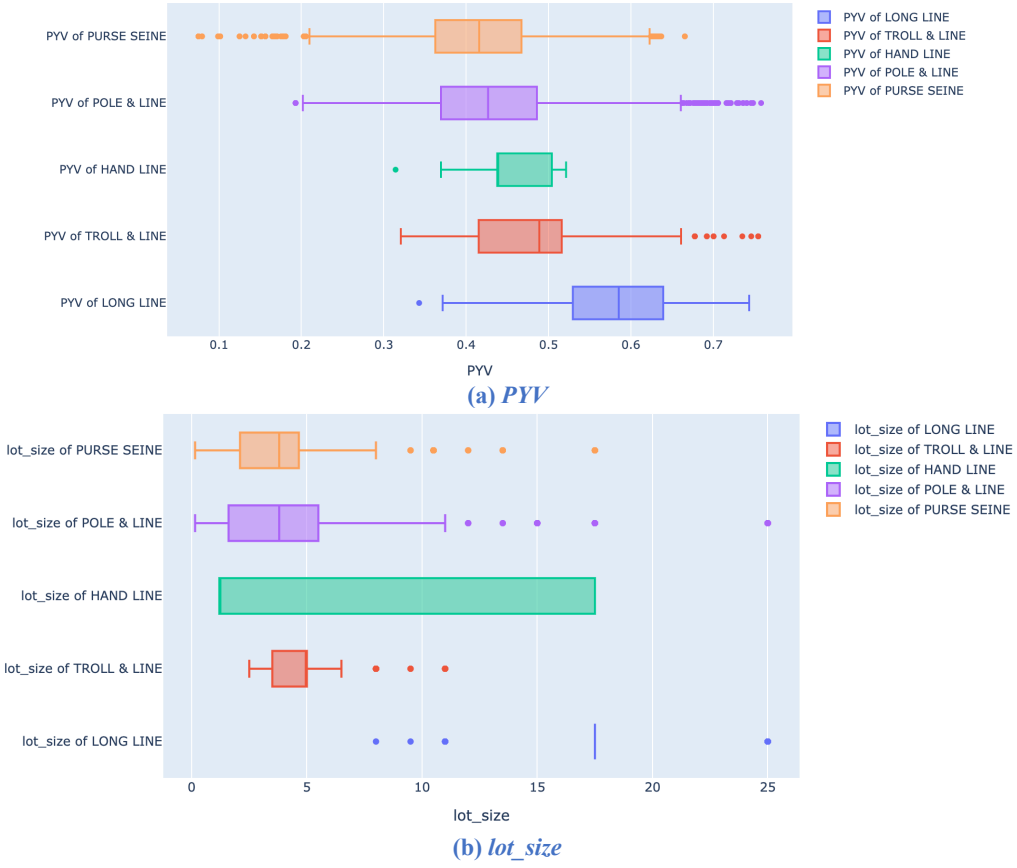


Figure 23 Summary Statistics of *lot_size* for Levels of *CT_Method* for All Fish Species

Therefore, we conclude that the highest mean value of *lot_size* is associated with 'LONG LINE' catching method whereas 'PURSE SEIN' catching method is associated with the lowest mean value of *lot_size*.

To analyze the impact of *lot_size* on *PYV*, we study Figure 22 and Figure 23, where we conclude that larger *lot_size* is associated with higher *PYV* across different levels of *CT_Method* except 'HAND LINE'. To better visualize this observation, we also study Box Plots of *lot_size* and *PYV* as shown in Figure 24. As can be seen, 'LONG LINE' catching method produces the highest *PYV* and largest *lot_size*, while 'PURSE SEINE' catching method is associated with the lowest *PYV* and smallest *lot_size.*



(a) *PYV*

(b) *lot_size*

Figure 24 Box Plots of (a) *PYV* and (b) *lot_size* for Levels of *CT_Method* for All Fish Species

5.4.1.4 Lasso Regression Coefficients

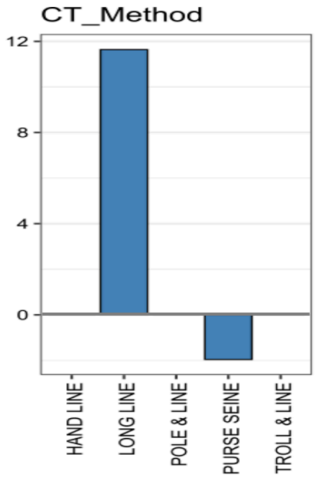


Figure 25 Lasso Regression Coefficients for Levels of *CT_Method* and *PYV*

To identify the impact of different levels of *CT_Method* on *PYV*, we used Lasso regression coefficients technique as shown in Figure 25.

It can be seen that 'LONG LINE' catching method has the highest positive impact, while 'PURSE SEINE' is associated with the highest negative impact and 'HAND LINE', 'POLE & LINE' and 'TROLL & LINE' are neutral on *PYV*.

5.4.2 *CT_Area*

CT_Area is the parameter that characterizes a particular area of fish catching with the levels like 'INDIAN OCEAN' or 'WESTERN PACIFIC' ocean.

5.4.2.1 ANOVA Test

We applied ANOVA test with the following null hypothesis:

H_0 =*PYV* for different levels of *CT_Area* are equal.

As a result of this test, *p*-value is zero which is less than the threshold value of 0.05. Accordingly, we rejected the above null hypothesis, i.e., *PYV* is different for different levels of *CT_Area* for all fish species.

5.4.2.2 Summary Statistics of *PYV* for All Fish Species

To study the variation of *PYV* across different levels of *CT_Area* for all fish species, we have computed summary statistics of *PYV*. These findings are summarized in Table 16 and visualized in Figure 26.

Table 16 Summary Statistic of *PYV* for Levels of *CT_Area* for All Fish Species

CT_Area	Mean value	Median value	STD
'EASTERN PACIFIC'	0.5243	0.5154	0.1145
'WESTERN PACIFIC'	0.4210	0.4200	0.0833
'NORTHWEST PACIFIC'	0.4701	0.4756	0.0899
'NORTHEAST PACIFIC'	0.4946	0.4868	0.0939
'SOUTHWEST PACIFIC'	0.4761	0.4754	0.0817
'INDIAN OCEAN'	0.4183	0.4168	0.0842
'JAVA SEA'	0.3619	0.3388	0.0903

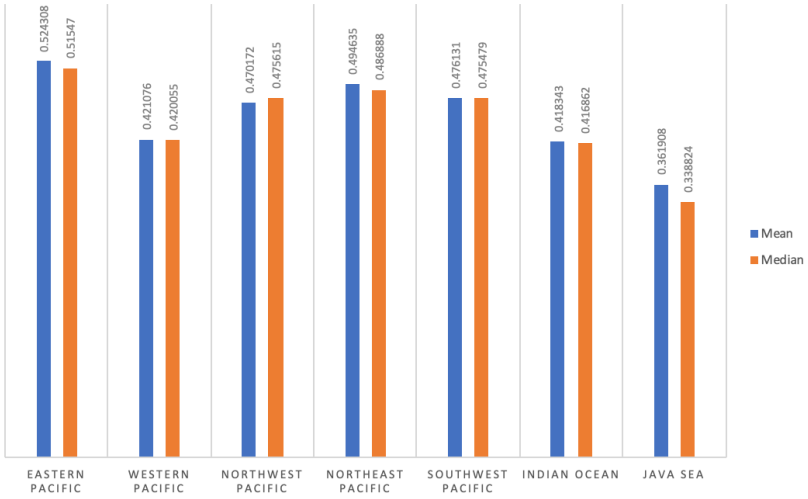


Figure 26 Summary Statistics of *PYV* for Levels of *CT_Area* for All Fish Species

It can be concluded that the highest mean value of *PYV* is associated with 'EASTERN PACIFIC' catching area while 'JAVA SEA' catching area is associated with the lowest mean value of *PYV*.

5.4.2.3 Summary Statistics of *lot_size* for All Fish Species

To study the variation of *lot_size* across levels of *CT_Area*, we have calculated summary statistics of *lot_size* for all fish species. Table 17 presents and Figure 27 visualizes these findings.

Table 17 Summary Statistics of *lot _size* for Levels of *CT_Area* for All Fish Species

CT_Area	Mean Value	Median Value	STD
'EASTERN PACIFIC'	14.6192	17.50	6.8459
'WESTERN PACIFIC'	4.1759	3.80	3.4184
'NORTHWEST PACIFIC'	6.1998	3.50	3.9870
'NORTHEAST PACIFIC'	6.8120	6.50	2.2901
'SOUTHWEST PACIFIC'	4.9999	5.00	1.2328
'INDIAN OCEAN'	4.2600	3.15	0.0842
'JAVA SEA'	2.1438	2.10	0.7712

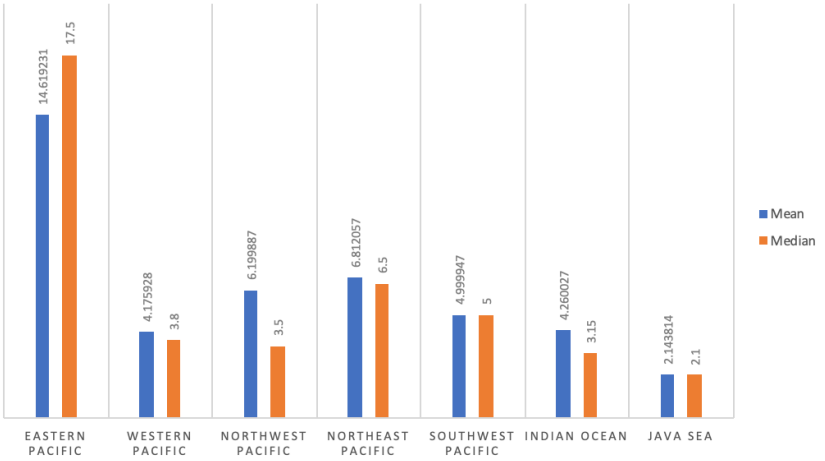
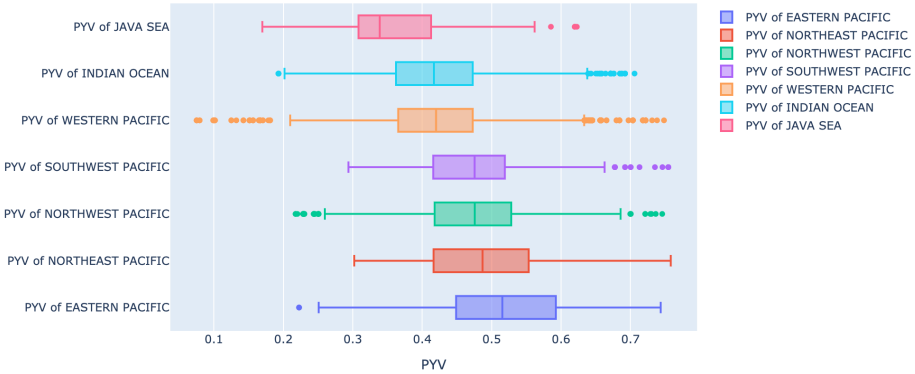


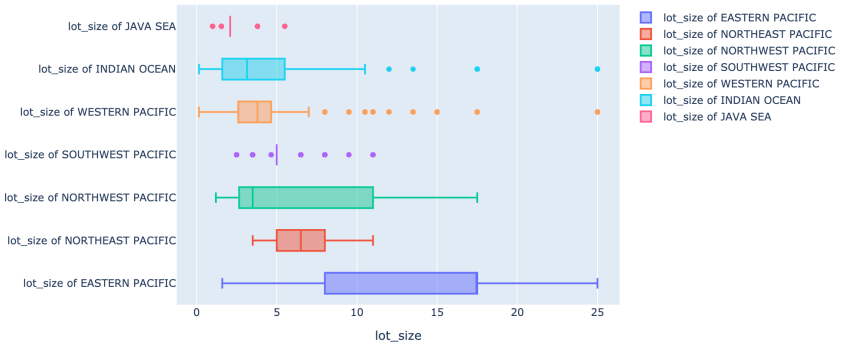
Figure 27 Summary Statistics of *lot_size* for Levels of *CT_Area* for All Fish Species

Therefore, we conclude that the highest mean value of *lot_size* is associated with 'EASTERN PACIFIC' catching area whereas 'JAVA SEA' catching area is associated with the lowest mean value of *lot_size*.

To analyze the impact of *lot_size* on *PYV*, we study Figure 26 and Figure 27, where we conclude that larger *lot_size* is associated with higher *PYV* across different levels of *CT_Area* except 'NORTH WEST'. To better visualize this observation, we also study Box Plots of *PYV* and *lot_size* as shown in Figure 28. As can be seen, 'EASTERN PACIFIC' catching area produces the highest *PYV* and largest *lot_size*, while 'JAVA SEA' catching area is associated with the lowest *PYV* and smallest *lot_size*.



(a) *PYV*



(b) *lot_size*

Figure 28 Box Plots of (a) *PYV* and (b) *lot_size* for Levels of *CT_Area* for All Fish Species

5.4.2.4 Lasso Regression Coefficients

To identify the impact of different levels of *CT_Area* on *PYV*, we used Lasso regression coefficients technique as shown in Figure 29.

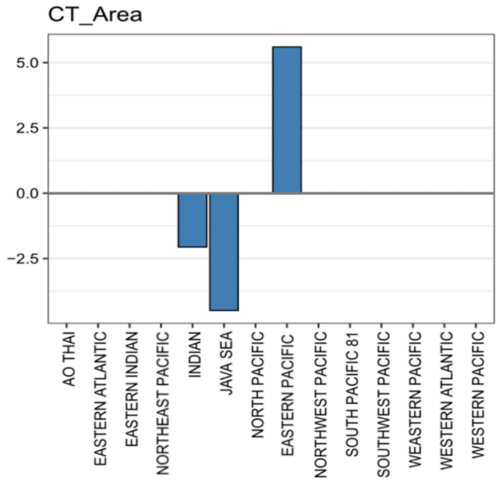


Figure 29 Lasso Regression Coefficients for Levels of *CT_Area* and *PYV*

It can be seen that 'NORTHEAST PACIFIC' catching area has the highest positive impact, while 'JAVA SEA' and 'INDIAN' are associated with the highest negative impacts and other fishing areas are neutral on *PYV*.

5.4.3 *FV_Flag*

FV_Flag is the parameter that characterizes a particular country to which the fishing vessel belongs with the levels like 'JAPAN' or 'UNITED STATES'.

5.4.3.1 ANOVA Test

We applied ANOVA test with the following null hypothesis:

H_0 = *PYV* for different levels of *FV_Flag* are equal.

As a result of this test, *p*-value is zero, which is less than the threshold value of 0.05. Accordingly, we rejected the above null hypothesis, i.e., *PYV* is different for different levels of *FV_Flag* for all fish species.

5.4.3.2 Summary Statistics of *PYV* for All Fish Species

To study the variation of *PYV* across different levels of *FV_Flag* for all fish species, we have computed summary statistics of *PYV*. The findings are presented in Table 18 and visualized in Figure 30.

Table 18 Summary Statistics of *PYV* for Levels of *FV_Flag* for All Fish Species

FV_Flag	Mean Value	Median Value	STD
'FIJI'	0.580	0.596	0.0793
'CHINA'	0.569	0.564	0.0913
'JAPAN'	0.471	0.478	0.0901
'NEW ZEALAND'	0.469	0.471	0.0838
'USA'	0.452	0.444	0.0823
'SOLOMON ISLANDS'	0.425	0.437	0.0153
'KIRIBATI'	0.424	0.428	0.0911
'KOREA'	0.423	0.425	0.0930
'TAIWAN'	0.422	0.422	0.0751
'MARSHALL IS'	0.421	0.423	0.0760
'MALDIVES'	0.418	0.416	0.0843
'NAURU'	0.418	0.435	0.077
'PNG '	0.416	0.412	0.063
'MICRONESIA'	0.413	0.413	0.080
'INDONESIA'	0.397	0.398	0.095

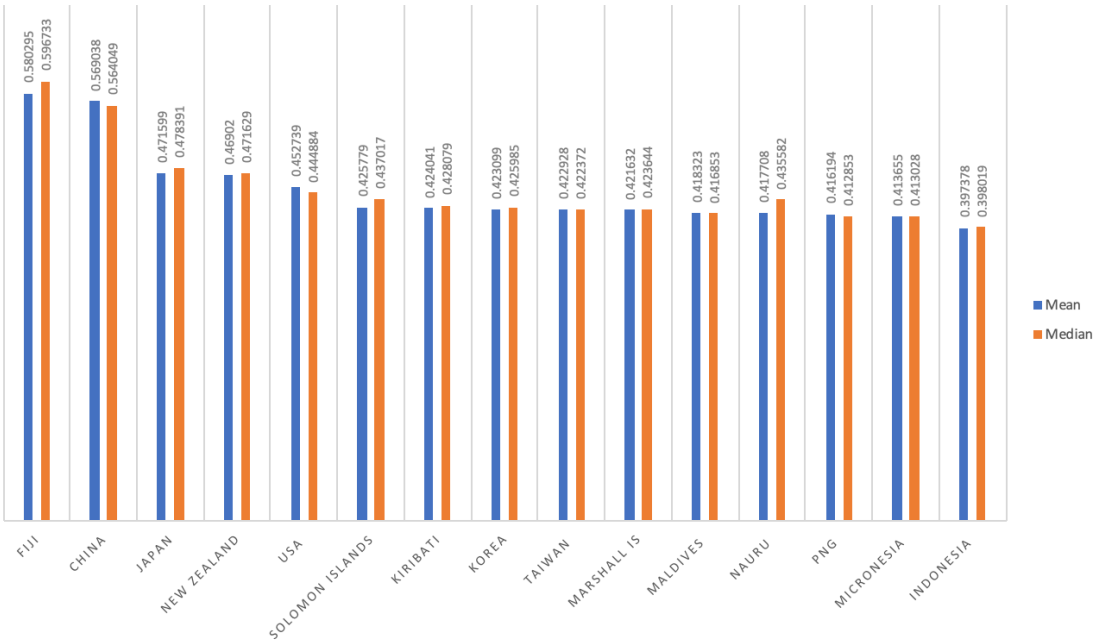


Figure 30 Summary Statistics of *PYV* for Levels of *FV_Flag* for All Fish Species

It can be concluded that the highest mean value of *PYV* is associated with 'FIJI' fishing vessel while 'INDONESIA' fishing vessel is associated with the lowest mean value of *PYV*.

5.4.3.3 Summary Statistics of *lot_size* for All Fish Species

To study the variation of *lot_size* across levels of *FV_Flag*, we have calculated summary statistics of *lot_size* for all fish species. The results are shown in Table 19 and depicted in Figure 31.

Table 19 Summary Statistics of *lot_size* for Levels of *FV_Flag* for All Fish Species

FV_Flag	Mean Value	Median Value	STD
'FIJI'	18.91	17.50	3.45
'CHINA'	17.18	17.50	6.44
'JAPAN'	6.37	3.65	4.01
'NEW ZEALAND'	4.95	5.00	1.45
'USA'	4.78	4.65	2.24
'SOLOMON ISLANDS'	17.50	17.50	0.00
'KIRIBATI'	3.45	2.65	1.27
'KOREA'	3.31	3.15	1.74
'TAIWAN'	3.98	3.80	1.42
'MARSHALL IS'	3.35	3.80	1.83
'MALDIVES'	4.25	3.15	3.66
'NAURU'	6.41	6.50	2.01
'PNG '	3.90	3.80	1.38
'MICRONESIA'	3.87	3.80	1.86
'INDONESIA'	4.17	2.10	5.55

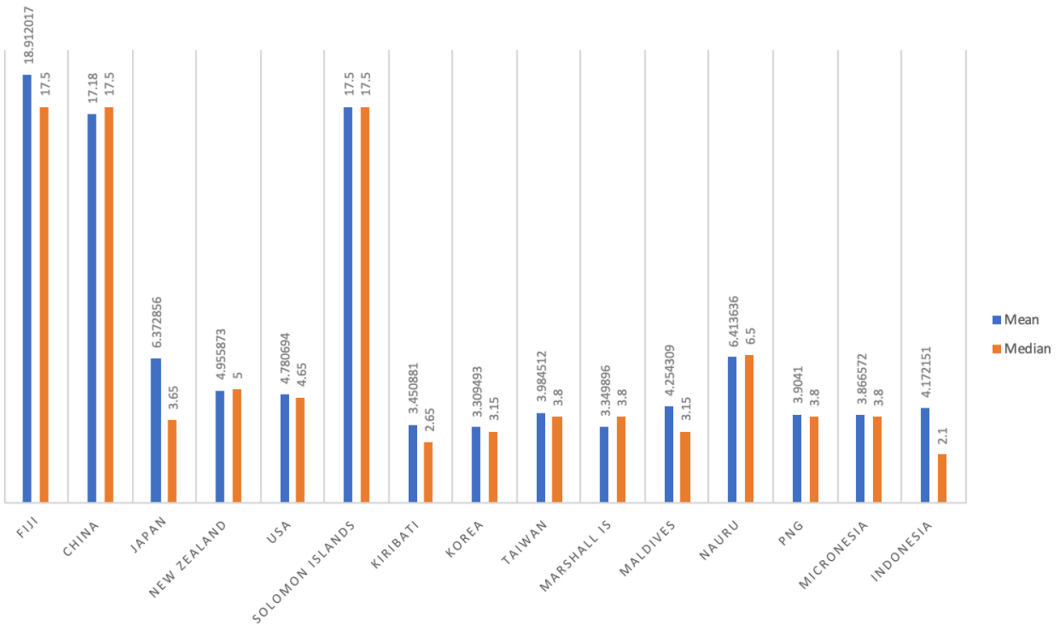
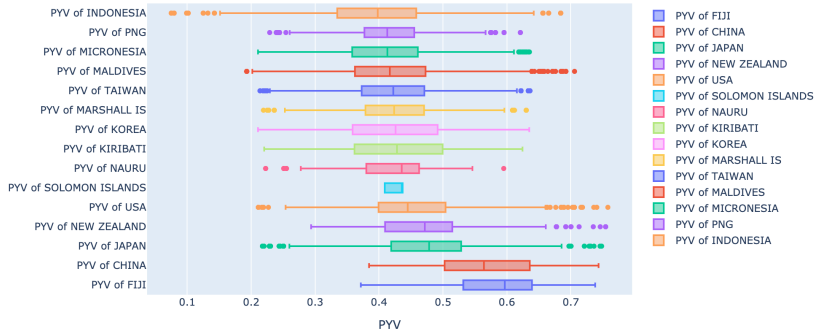


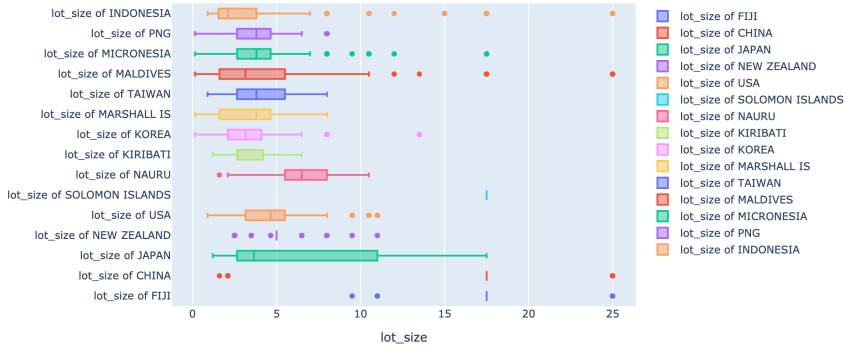
Figure 31 Summary Statistics of *Lot Size* for Levels of *FV_Flag* for All Fish Species

Therefore, we conclude that the highest mean value of *lot_size* is associated with 'FIJI' fishing vessel whereas 'KOREA' fishing vessel is associated with the lowest mean value of *lot_size*.

To analyze the impact of *lot_size* on *PYV*, we study Figure 30 and Figure 31, where we conclude that larger *lot_size* are not necessarily associated with higher *PYV* across different levels of *FV_Flag,* e.g., 'FIJI' fishing vessel produces the highest *PYV* and largest *lot_size*, while 'JAPAN' fishing vessel, despite having large *lot_size*, has produced small *PYV* (see Figure 32).



(a) *PYV*



(b) *lot_size*

Figure 32 Box Plots of (a) *PYV* and (b) lot_size for Levels of *FV_Flag* for All Fish Species

5.4.3.4 Lasso Regression Coefficients

To identify the impact of different levels of *FV_Flag* on *PYV*, we used Lasso regression coefficients technique as shown in Figure 33.

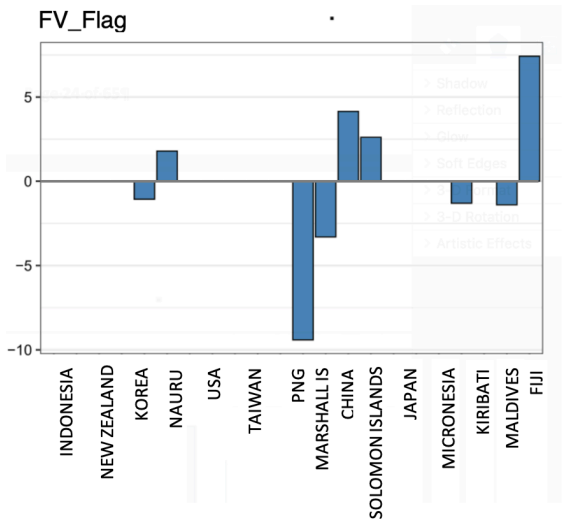


Figure 33 Lasso Regression Coefficients for Levels of *FV_Flag* and *PYV*

It can be seen that 'FIJI' fishing vessel has the highest positive impact, while 'PNG' fishing vessel is associated with the highest negative impacts on *PYV*.

5.5 Data Analysis of Raw Material Parameters Stratified for Fish Species

The same study as performed in Section 0, along with some additional tests, has been conducted for stratified Working dataset for each fish species. The results are presented in coming subsections.

5.5.1 *CT_Method*

In addition to ANOVA Test and Statistics Summary Calculation of *PYV* and *lot_size*, we have applied One-Tailed Independent *t*-test and Point-Biserial Correlation Coefficient techniques for analyzing *PYV*.

5.5.1.1 ANOVA Test

Table 20 summarizes the findings.

Table 20 *p*-values of ANOVA Test for *CT_Method* of Each Fish Species

Fish species	AL	SK	YF	BE	BT	TG
***p*-value**	7.0e-89	1.90e-20	0.43e-10	N/A*	N/A*	N/A**

* BE and BT have only two levels of *CT_Method* which shall be studied in next subsection.
** TG has only one level of *CT_Method*.

As a result of this test, *p*-values are less than the threshold value of 0.05. Accordingly, we rejected the null hypothesis, i.e., *PYV* is different across different levels of *CT_Method* for each fish species.

5.5.1.2 One-Tailed Independent *t*-test

We apply One-Tailed Independent *t*-test to order the levels of catching methods according to their impact on *PYV* based on the H_0 hypothesis. The results are shown in Table 21.

Table 21 One-Tailed Independent t-test Results of *CT_Method* for Each Fish Species

Fish Species	H_0 Hypothesis	Statistic value	*p*-value	Fail to Reject / Reject
AL	'POLE & LINE' <= 'TROLL & LINE'	2.90	3.80e-3	Reject
	'POLE & LINE' <= 'LONG LINE'	-20.03	2.74e-84	Fail to Reject
	'TROLL & LINE' <= 'LONG LINE'	-18.70	6.61e-65	Fail to Reject
SK	'POLE & LINE' <= 'PURSE SEINE'	-9.50	2.14e-21	Fail to Reject
	'POLE & LINE' <= 'HAND LINE'	-0.87	3.85e-1	Fail to Reject
	'PURSE SEINE' <= 'HAND LINE'	-0.63	5.31e-1	Fail to Reject
YF	'POLE & LINE' <= 'PURSE SEINE'	-0.59	5.55e-1	Fail to Reject
	'POLE & LINE' <= 'HAND LINE'	-1.15	2.49e-1	Fail to Reject
	'PURSE SEINE' <= 'HAND LINE'	0.72	4.75e-1	Fail to Reject
BE	'POLE & LINE' <= 'PURSE SEINE'	0.31	7.59e-1	Fail to Reject
BT	'POLE & LINE' <= 'PURSE SEINE'	4.22	2.57e-05	Reject
TG	'PURSE SEINE'	N/A*	N/A*	N/A*

* TG has only one level of *CT_Method*.

According to the above table, the ordering of *PYV* across different levels of *CT_Method* for each fish species is presented in Table 22.

Table 22 Order of *PYV* across *CT_Method* Levels for Each Fish Species Based on One-Tailed Independent *t*-test

Fish Species	Order of *CT_Method* Levels
AL	'TROLL & LINE' <= 'POLE & LINE' <= 'LONGE LINE'
SK	'POLE & LINE' <= 'PURSE SEINE' <= 'HAND LINE'
YF	'POLE & LINE' <= 'PURSE SEINE' <= 'HAND LINE'
BE	'POLE & LINE' <= 'PURSE SEINE'
BT	'PURSE SEINE' <= 'POLE & LINE'
TG	'PURSE SEINE'

5.5.1.3 Summary Statistics of *PYV* for Each Fish Species

Table 23 shows and Figure 34 visualizes the findings.

Table 23 Summary Statistics of *PYV* for Levels of *CT_Method* for Each Fish Species

Fish Species	*CT_Method*	Mean value	Median value	STD
AL	'LONG LINE'	0.58	0.59	8.28e-2
	'POLE & LINE'	0.49	0.49	8.28e-2
	'TROLL & LINE'	0.48	0.49	8.69e-2
SK	'POLE & LINE'	0.42	0.41	8.38e-2
	'HAND LINE'	0.44	0.44	0.00
	'PURSE SEINE'	0.42	0.42	7.71e-2
YF	'POLE & LINE'	0.44	0.43	8.87e-2
	'HAND LINE'	0.46	0.51	8.41e-2
	'PURSE SEINE'	0.45	0.46	6.87e-2
BE	'POLE & LINE'	0.42	0.46	6.39e-2
	'PURSE SEINE'	0.48	0.50	6.57e-2
BT	'POLE & LINE'	0.48	0.52	0.12
	'PURSE SEINE'	0.36	0.36	8.98e-2
TG	'PURSE SEINE'	0.43	0.42	8.15e-2

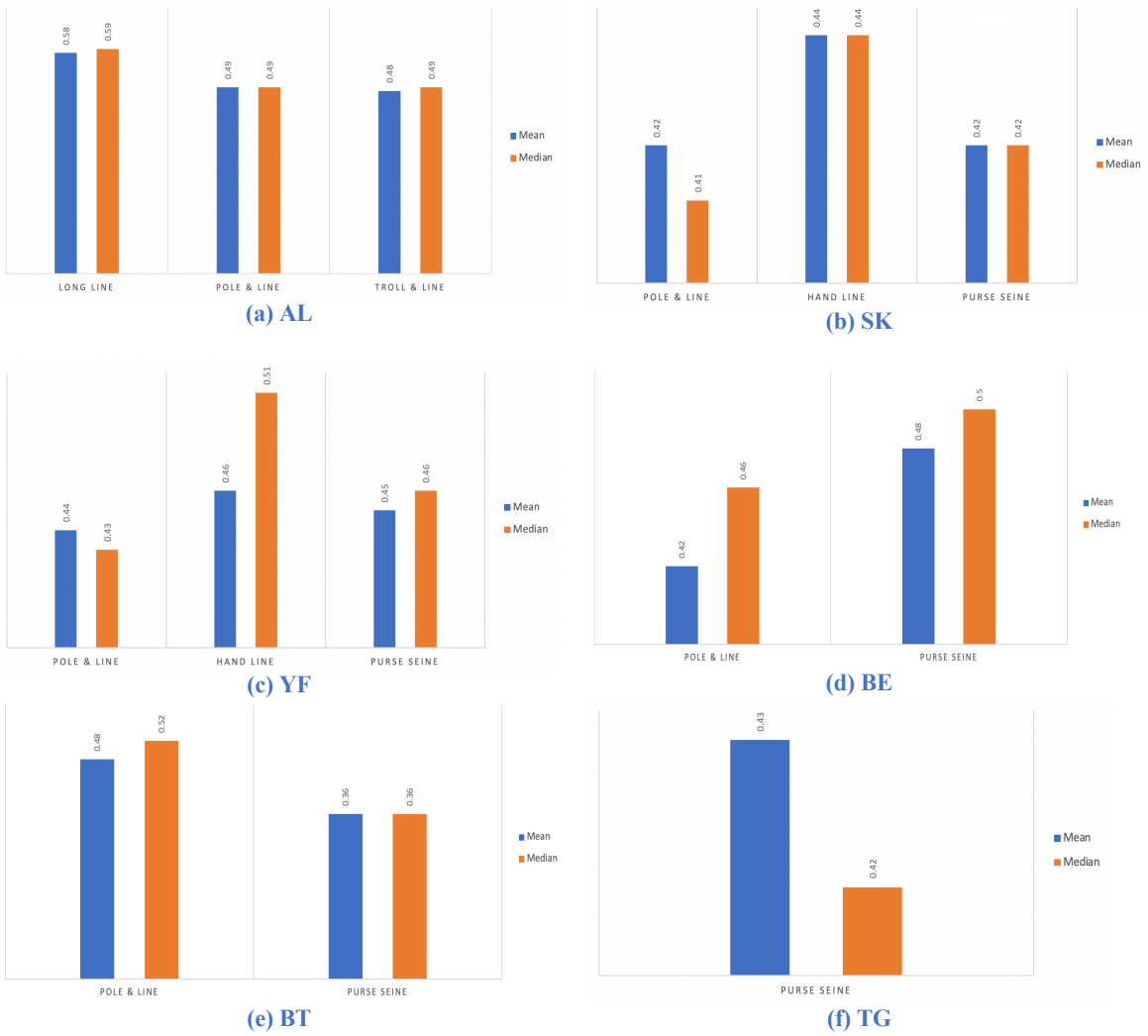


Figure 34 Summary Statistics of *PYV* for Levels of *CT_Method* for Each Fish Species

Based on the results shown in Table 23 and Figure 34, the ordering of *PYV* across different levels of *CT_Method* for each fish species is presented in Table 24.

Table 24 Order of *PYV* across *CT_Method* Levels for Each Fish Species Based on Summary Statistics

Fish Species	Order of *CT_Method* Levels
AL	'TROLL & LINE' <= 'POLE & LINE' <= 'LONG LINE'
SK	'POLE & LINE' <= 'PURSE SEINE' <= 'HAND LINE'
YF	'POLE & LINE' <= 'PURSE SEINE' <= 'HAND LINE'
BE	'POLE & LINE' <= 'PURSE SEINE'
BT	'PURSE SEINE' <= 'POLE & LINE'
TG	'PURSE SEINE'

5.5.1.4 Summary Statistics of *lot_size* for Each Fish Species

The findings are summarized in Table 25 and shown in Figure 35.

Table 25 Summary Statistics of *lot_size* for Levels of *CT_Method* for Each Fish Species

Fish Species	*CT_Method*	Mean value	Median value	STD
AL	'LONG LINE'	19.04	17.50	3.41
	'POLE & LINE'	7.32	5.00	2.89
	'TROLL & LINE'	5.04	5.00	1.85
SK	'HAND LINE'	3.79	3.15	2.13
	'POLE & LINE'	1.20	1.20	0.00
	'PURSE SEINE'	3.75	3.80	1.67
YF	'POLE & LINE'	6.92	2.10	7.55
	'HAND LINE'	17.50	17.50	0.00
	'PURSE SEINE'	8.21	8.00	5.87
BE	'POLE & LINE'	4.82	5.50	0.93
	'PURSE SEINE'	4.56	3.80	2.64
BT	'POLE & LINE'	2.19	2.10	0.92
	'PURSE SEINE'	4.82	5.50	0.93
TG	'PURSE SEINE'	2.54	2.60	1.05

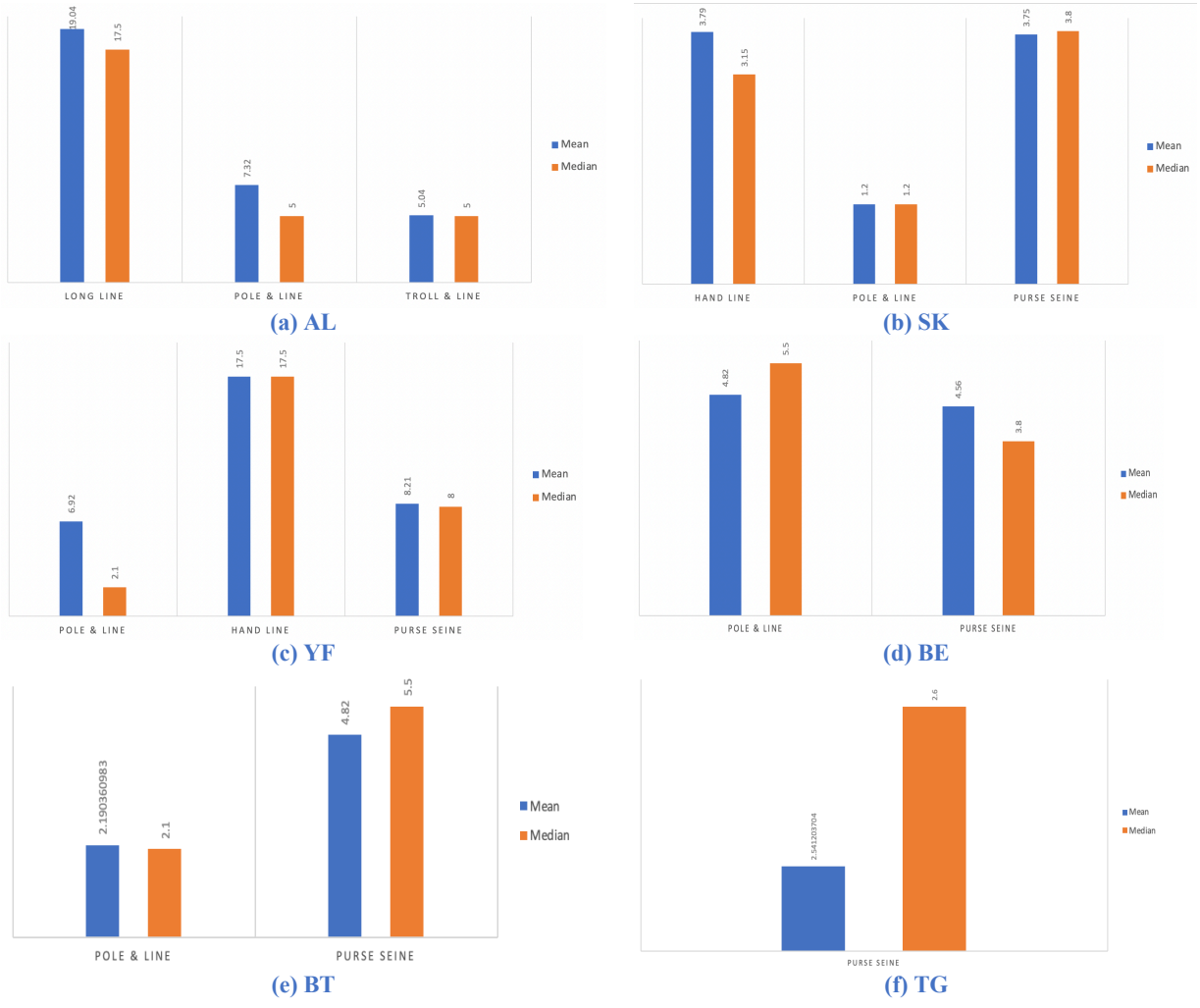


Figure 35 Summary Statistics of *Lot Sizes* for Levels of *CT_Method* for Each Fish Species

Based on the results shown in Table 25 and Figure 35, the ordering of *lot_size* across different levels of *CT_Method* for each fish species is presented in Table 26.

Table 26 Order of *lot_size* across *CT_Method* Levels for Each Fish Species Based on Summary Statistics

Fish Species	Order of *CT_Method* Levels
AL	'TROLL & LINE' <= 'POLE & LINE' <= 'LONG LINE'
SK	'POLE & LINE' <= 'PURSE SEINE' <= 'HAND LINE'
YF	'POLE & LINE' <= 'PURSE SEINE' <= 'HAND LINE'
BE	'PURSE SEINE' <= 'POLE & LINE'
BT	'POLE & LINE' <= 'PURSE SEINE'
TG	'PURSE SEINE'

Based on Figure 34 and Figure 35, we conclude that larger *lot_size* is associated with higher *PYV* across different levels of *CT_Method* for each fish species except for BE and BT species.

5.5.1.5 Point-Biserial Correlation Coefficients

In order to study the correlation between *PYV* and different levels of *CT_Method* for each fish species, we computed the point-biserial correlation coefficients. The findings are shown in Table 27.

Table 27 Point-Biserial Correlation Coefficients of *PYV* and Levels of *CT_Method* for Each Fish Species

Fish Species	*CT_Method*	Correlation Value
AL	'TROLL & LINE'	-0.074
	'POLE & LINE'	-2.16e-01
	'LONG LINE'	3.50e-01
SK	'POLE & LINE'	-6.16e-02
	'PURSE SEINE'	4.97e-3
	'HAND LINE'	6.14e-02
YF	'POLE & LINE'	-0.015
	'HAND LINE'	0.017
	'PURSE SEINE'	8.86e-3
BE	'POLE & LINE'	-0.018
	'PURSE SEINE'	0.018
BT	'POLE & LINE'	0.11
	'PURSE SEINE'	-0.11
TG	N/A*	N/A*

* TG has only one level of *CT_Method*.

According to Table 27, the order of Point-Biserial Correlation Coefficients of *PYV* across different levels of *CT_Method* is summarized in Figure 36.

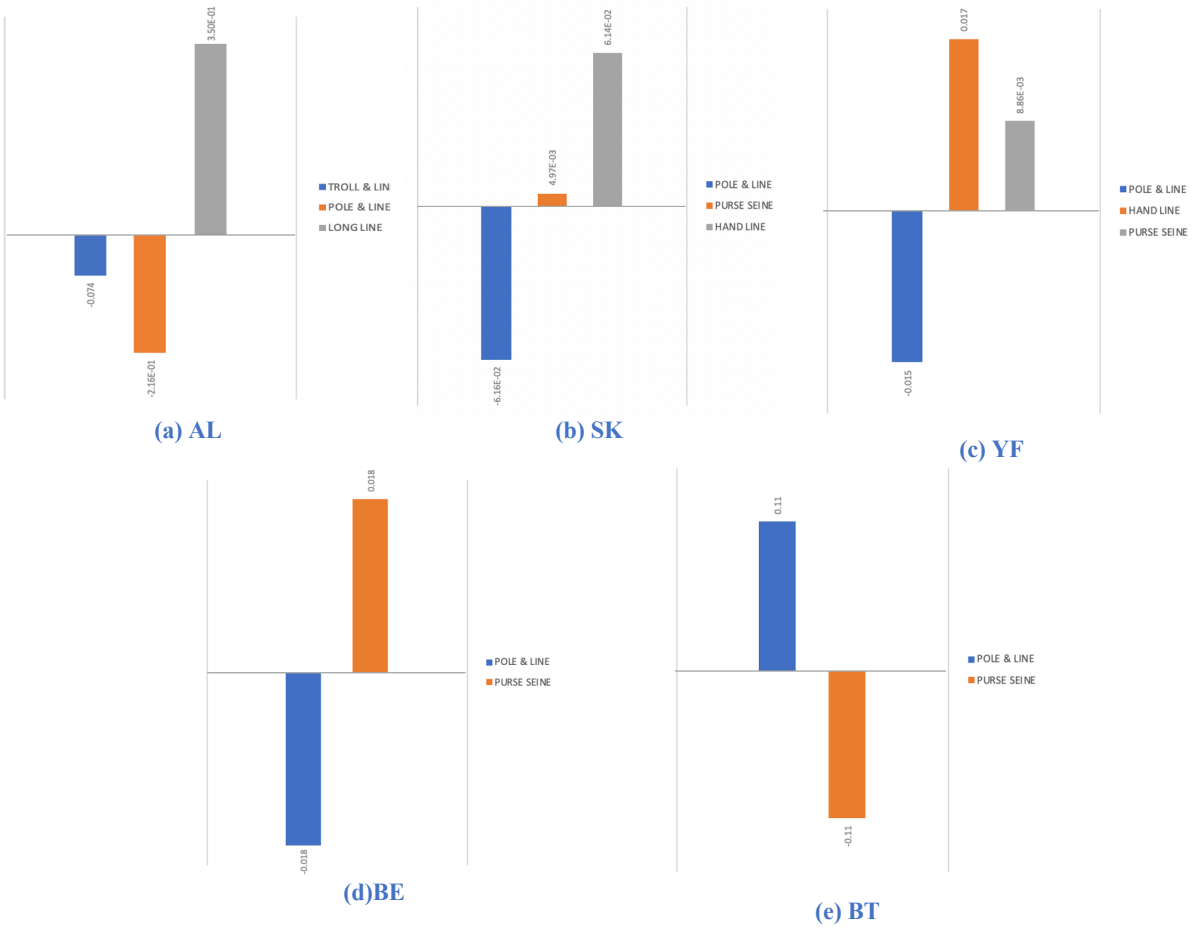


Figure 36 Point-Biserial Correlation Coefficients of *PYV* Across Levels of *CT_Method* for Each Fish Species

5.5.2 *CT_Area*

In addition to ANOVA Test and Statistics Summary Calculation of *PYV* and *lot_size*, we have applied One-Tailed Independent *t*-test and Point-Biserial Correlation Coefficient techniques for analyzing *PYV*.

5.5.2.1 ANOVA Test

Table 28 shows the results.

Table 28 *p*-values of ANOVA Test for *CT_Area* of Each Fish Species

Fish Species	AL	SK	YF	BE	BT	TG
***p*-value**	2.14e-96	1.17e-48	N/A*	N/A*	1.46e-06	N/A*

* YF, BE, and TG have only two levels of *CT_Area* which shall be studied in next subsection.

As a result of this test, *p*-values are less than the threshold value of 0.05. Accordingly, we rejected the null hypothesis, i.e., *PYV* is different across different levels of *CT_Area* for each fish species.

5.5.2.2 One-Tailed Independent t-test

We apply One-Tailed Independent *t*-test to order the levels of catching areas according to their impact on *PYV* based on the H_0 hypothesis. The findings are shown in Table 29.

Table 29 One-Tailed Independent *t*-test Results for *CT_Area* of Each Fish Species

Fish Species	H_0 Hypothesis	Statistic value	*p*-value	Fail to Reject / Reject
AL	'EASTERN PACIFIC' <= 'SOUTHWEST PACIFIC'	13.11	2.73e-37	Reject
	'EASTERN PACIFIC' <= 'NORTHWEST PACIFIC'	9.89	6.67e-22	Reject
	'EASTERN PACIFIC' <= 'WESTERN PACIFIC'	-0.035	0.97	Fail to Reject
	'EASTERN PACIFIC' <= 'NORTHEAST PACIFIC'	11.42	1.77e-28	Reject
	'SOUTHWEST PACIFIC' <= 'NORTHWEST PACIFIC'	-6.30	3.64e-10	Fail to Reject
	'SOUTHWEST PACIFIC' <= 'WESTERN PACIFIC'	-18.41	3.70e-69	Fail to Reject
	'SOUTHWEST PACIFIC' <= 'NORTHEAST PACIFIC'	-0.08	0.93	Fail to Reject
	'NORTHWEST PACIFIC' <= 'WESTERN PACIFIC'	-13.12	1.75e-36	Fail to Reject
	'NORTHWEST PACIFIC' <= 'NORTHEAST PACIFIC'	5.27	1.59e-07	Reject
	'WESTERN PACIFIC' <= 'NORTHEAST PACIFIC'	15.80	4.58e-51	Reject
SK	'EASTERN PACIFIC' <= 'INDIAN OCEAN'	1.91	0.056	Reject
	'EASTERN PACIFIC' <= 'NORTHWEST PACIC'	-0.85	0.39	Fail to Reject
	'EASTERN PACIFIC' <= 'WESTERN PACIFIC'	0.67	0.50	Fail to Reject
	'INDIAN OCEAN' <= 'NORTHWEST PACIFIC'	-8.40	5.21e-17	Fail to Reject
	'INDIAN OCEAN' <= 'WESTERN PACIFIC'	-13.31	2.52e-40	Fail to Reject
	'NORTHWEST PACIFIC' <= 'WESTERN PACIFIC'	4.93	8.20e-07	Reject
YF	'WESTERN PACIFIC' <= 'INDIAN OCEAN'	1.04	0.30	Fail to Reject
BE	'WESTERN PACIFIC' <= 'INDIAN OCEAN'	-0.31	0.76	Fail to Reject
BT	'WESTERN PACIFIC' <= 'JAVA SEA'	3.04	2.41e-3	Reject
	'WESTERN PACIFIC' <= 'INDIAN OCEAN'	-4.05	5.44e-05	Fail to Reject
	'JAVA SEA' <= 'INDIAN OCEAN'	-4.71	3.65e-06	Fail to Reject
TG	'WESTERN PACIFIC' <= 'JAVA SEA'	3.70	2.46e-4	Reject

According to the above table, the ordering of *PYV* across different levels of *CT_Area* for each fish species is presented in Table 30.

Table 30 Order of *PYV* across Levels of *CT_Area* for Each Fish Species Based on One-Tailed Independent *t*-test

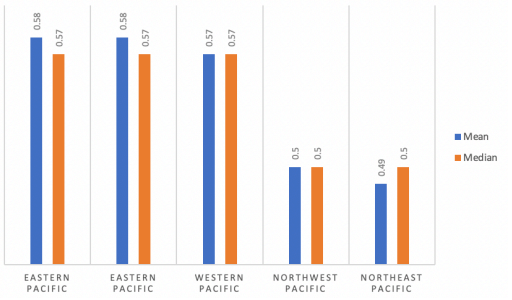
Fish Species	Order of *CT-Area* Levels
AL	'NORTHEAST PACIFIC' <= 'SOUTHWEST PACIFIC' <= 'NORTHWEST PACIFIC' <= 'WESTERN PACIFIC' <= 'EASTERN PACIFIC'
SK	'EASTERN PACIFIC' <= 'INDIAN OCEAN' <= 'WESTERN PACIFIC' <= 'NORTHWEST PACIFIC'
YF	'WESTERN PACIFIC' <= 'INDIAN OCEAN'
BE	'WESTERN PACIFIC' <= 'INDIAN OCEAN'
BT	'JAVA SEA' <= 'WESTERN PACIFIC' <= 'INDIAN OCEAN'
TG	'JAVA SEA' <= 'WESTERN PACIFIC'

5.5.2.3 Summary Statistics of *PYV* for Each Fish Species

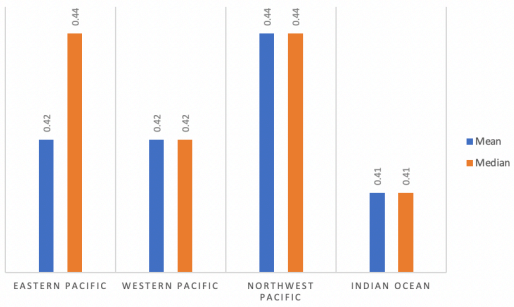
Table 31 summarizes and Figure 37 depicts the results.

Table 31 Summary Statistics of PYV for Levels of CT_Area for Each Fish Species

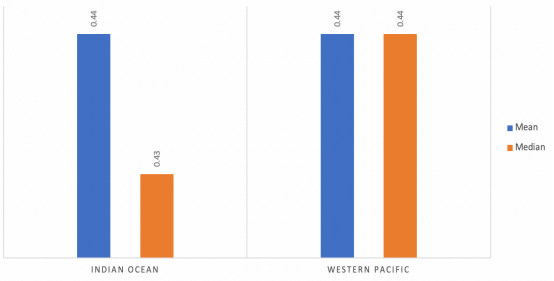
Fish Species	*CT_Area*	Mean value	Median value	STD
AL	'EASTERN PACIFIC'	0.58	0.57	8.62e-2
	'WESTERN PACIFIC'	0.57	0.57	8.60e-2
	'NORTHWEST PACIFIC'	0.50	0.50	8.01e-2
	'NORTHEAST PACIFIC'	0.49	0.50	9.40e-2
	'SOUTHWEST PACIFIC'	0.48	0.47	8.17e-2
SK	'EASTERN PACIFIC'	0.42	0.44	7.72e-2
	'WESTERN PACIFIC'	0.42	0.42	7.72e-2
	'NORTHWEST PACIFIC'	0.44	0.44	8.95e-2
	'INDIAN OCEAN'	0.41	0.41	8.32e-2
YF	'INDIAN OCEAN'	0.44	0.43	8.61e-2
	'WESTERN PACIFIC'	0.44	0.44	0.10
BE	'INDIAN OCEAN'	0.46	0.50	6.57e-2
	'WESTERN PACIFIC'	0.41	0.46	6.39e-2
BT	'INDIAN OCEAN'	0.48	0.52	0.12
	'WESTERN PACIFIC'	0.37	0.36	9.03e-2
	'JAVA SEA'	0.35	0.33	8.64e-2
TG	'JAVA SEA'	0.42	0.41	8.55e-2
	'WESTERN PACIFIC'	0.43	0.42	8.05e-2



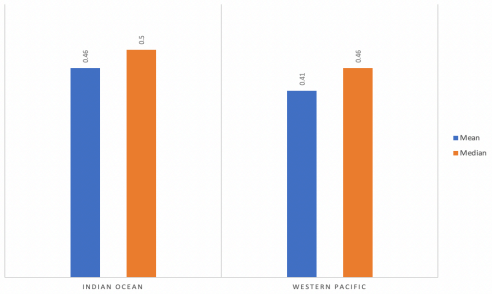
(a) AL



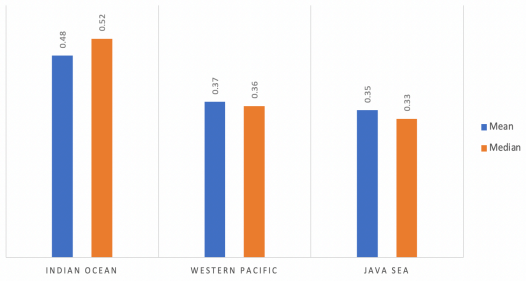
(b) SK



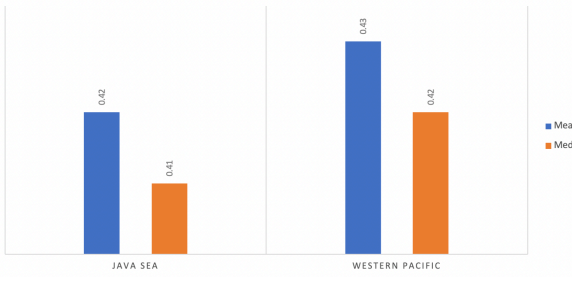
(c) YF



(d) BE



(e) BT



(f) TG

Figure 37 Summary Statistics of *PYV* for Levels of *CT_Area* for Each Fish Species

Based on the results shown in Table 31 and Figure 37, the ordering of *PYV* across different levels of *CT_Area* for each fish species is presented in Table 32.

Table 32 Order of *PYV* across *CT_Area* Levels for Each Fish Species Based on Summary Statistics

Fish Species	Order of the *CT_Area* Levels
AL	'SOUTHWEST PACIFIC' <= 'NORTHEAST PACIFIC'<='NORTHWEST PACIFIC'<= 'WESTERN PACIFIC' <= 'EASTERN PACIFIC'
SK	'EASTERN PACIFIC' <= 'INDIAN OCEAN'<= 'WESTERN PACIFIC' <= 'NORTHWEST PACIFIC'
YF	'WESTERN PACIFIC' <= 'INDIAN OCEAN'
BE	'WESTERN PACIFIC' <= 'INDIAN OCEAN'
BT	'JAVA SEA' <= 'WESTERN PACIFIC' <= 'INDIAN OCEAN'
TG	'JAVA SEA' <= 'WESTERN PACIFIC'

5.5.2.4 Summary Statistics of *lot_size* for Each Fish Species

The findings are summarized in Table 33 and shown in Figure 38.

Table 33 Summary Statistics of *lot_size* for Levels of *CT_Area* for Each Fish Species

Fish Species	CT Area	Mean value	Median value	STD
AL	'EASTERN PACIFIC'	19.31	17.5	3.22
	'WESTERN PACIFIC'	17.47	17.5	4.40
	'NORTHWEST PACIFIC'	9.52	11.0	2.64
	'NORTHEAST PACIFIC'	6.81	6.5	2.30
	'SOUTHWEST PACIFIC'	5.00	5.0	1.23
SK	'NORTHWEST PACIFIC'	6.39	6.5	2.04
	'WESTERN PACIFIC'	3.73	3.8	1.67
	'EASTERN PACIFIC'	2.54	2.65	0.38
	'INDIAN OCEAN'	3.87	3.8	2.18
YF	'WESTERN PACIFIC'	5.86	2.1	6.75
	'INDIAN OCEAN'	13.18	15.0	8.64
BE	'INDIAN OCEAN'	4.82	5.5	0.93
	'WESTERN PACIFIC'	4.56	3.8	2.64
BT	'INDIAN OCEAN'	2.27	1.55	1.45
	'WESTERN PACIFIC'	2.24	2.1	0.99
	'JAVA SEA'	1.97	2.1	0.47
TG	'WESTERN PACIFIC'	2.40	2.6	0.95
	'JAVA SEA'	3.03	2.1	1.27

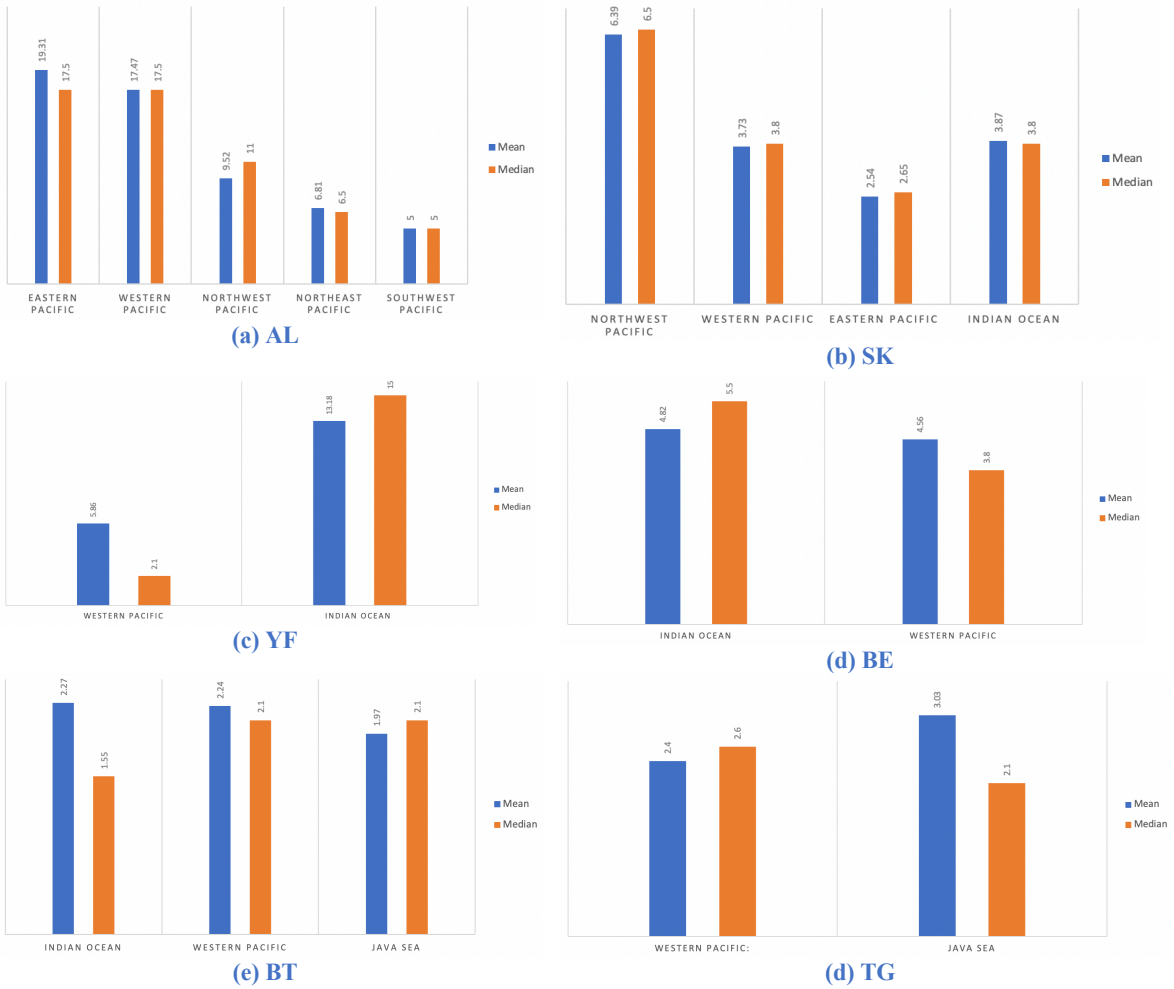


Figure 38 Summary Statistics of *Lot_ Sizes* for Levels of *CT_Area* for Each Fish Species

Based on the results shown in Table 33 and Figure 38, the ordering of *lot_size* across different levels of *CT_Area* for each fish species is presented in Table 34.

Table 34 Order of *lot_size* across *CT_Area* Levels for Each Fish Species Based on Summary Statistics

Fish Species	Order of *CT_Area* Levels
AL	'SOUTHWEST PACIFIC' <= 'NORTHEAST PACIFIC' <= 'NORTHWEST PACIFIC' <= 'WESTERN PACIFIC' <= 'EASTERN PACIFIC'
SK	'EASTERN PACIFIC' <= 'WESTERN PACIFIC' <= 'INDIAN OCEAN' <= 'NORTHWEST PACIFIC'
YF	'WESTERN PACIFIC' <= 'INDIAN OCEAN'
BE	'WESTERN PACIFIC' <= 'INDIAN OCEAN'
BT	'JAVA SEA' <= 'WESTERN PACIFIC' <= 'INDIAN OCEAN'
TG	'WESTERN PACIFIC' <= 'JAVA SEA'

Based on Table 32 and Table 34, we conclude that larger *lot_size* is associated with higher *PYV* across different levels of *CT_Method* for each fish species except for SK, BE and TG species.

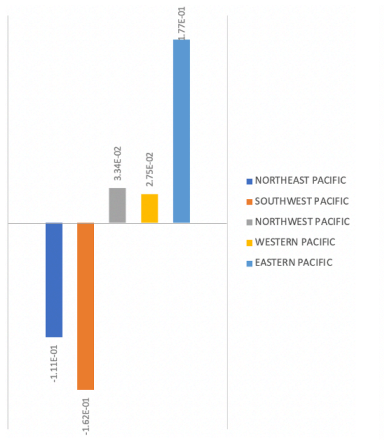
5.5.2.5 Point-Biserial Correlation Coefficients

In order to study the correlation between *PYV* and different levels of *CT_Area* for each fish species, we computed the point-biserial correlation coefficients. The results have been presented in Table 35.

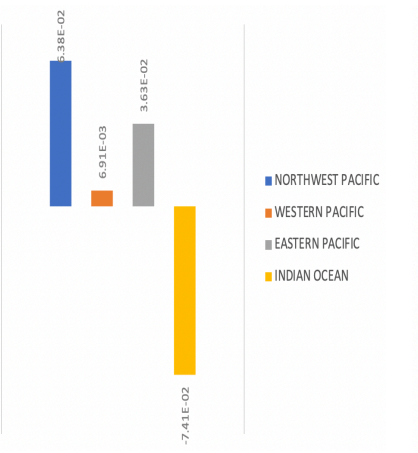
Table 35 Point-Biserial Correlation Coefficients of *PYV* and *CT_Area* for Each Fish Species

Fish Species	*CT_Area*	Correlation Value
AL	'NORTHEAST PACIFIC'	-1.11e-01
	'SOUTHWEST PACIFIC'	-1.62e-01
	'NORTHWEST PACIFIC'	3.34e-2
	'WESTERN PACIFIC'	2.75e-02
	'EASTERN PACIFIC'	1.77e-01
SK	'NORTHWEST PACIFIC'	6.38e-02
	'WESTERN PACIFIC'	6.91e-3
	'EASTERN PACIFIC'	3.63e-02
	'INDIAN OCEAN'	-7.41e-02
YF	'INDIAN OCEAN'	1.57e-2
	'WESTERN PACIFIC'	-1.57e-2
BE	INDIAN OCEAN:	1.81e-2
	WESTERN PACIFIC:	-1.80e-2
BT	INDIAN OCEAN:	0.11
	WESTERN PACIFIC:	6.19e-2
	JAVA SEA	-8.60e-2
TG	WESTERN PACIFIC:	0.18
	JAVA SEA	-0.18

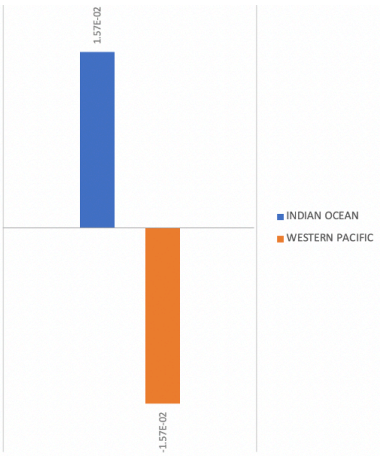
According to Table 35, the order of Point-Biserial Correlation Coefficients of *PYV* across different levels of *CT_Area* is summarized in Figure 39.



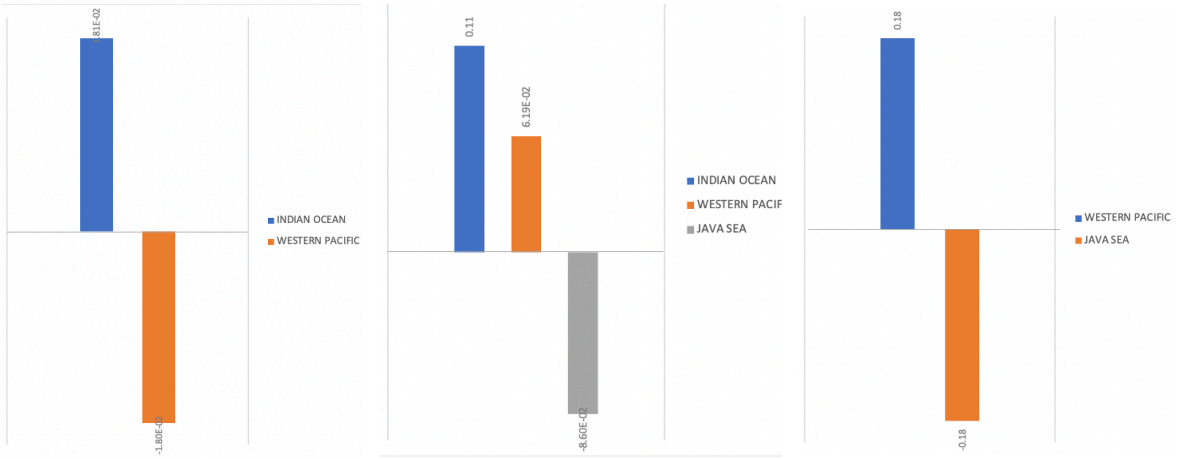
(a) AL



(b) SK



(c)YF



d) BE (e) BT (f) TG

Figure 39 Point-Biserial Correlation Coefficients of *PYV* Across Levels of *CT_Area* for Each Fish Species

5.5.3 *FV_Flag*

In addition to ANOVA Test and Statistics Summary values of *PYV* and *lot_size*, we applied One-Tailed Independent *t*-test and Point-Biserial Correlation Coefficient techniques for analyzing *PYV*.

5.5.3.1 ANOVA Test

Table 36 summarizes the results.

Table 36 *p*-values of ANOVA Test for *FV_Flag* of Each Fish Species

Fish Species	AL	SK	YF	BE	BT	TG
***p*-value**	2.22e-89	8.20e-27	9.1e-26	2.10e-4	0.10e-4	N/A*

* TG has only one level of *FV-Flag*.

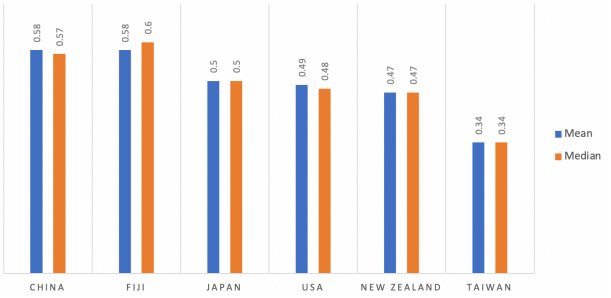
As a result of this test, *p*-values are less than the threshold value of 0.05. Accordingly, we rejected the null hypothesis, i.e., *PYV* is different across different levels of *FV_Flag* for each fish species.

5.5.3.2 Summary Statistics of *PYV* for Each Fish Species

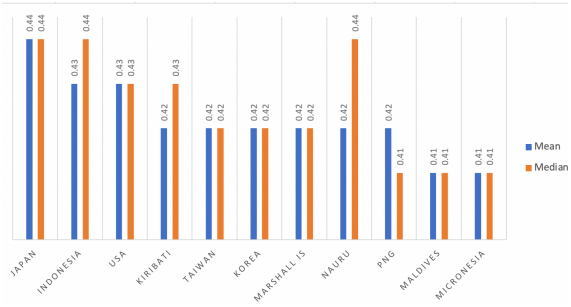
Table 37 summarizes and Figure 40 shows the results.

Table 37 Summary Statistics of *PYV* for Levels of *FV_Flag* for Each Fish Species

Fish Species	*FV_Flag*	Mean Value	Median Value	STD
AL	'CHINA'	0.58	0.57	0.09
	'FIJI'	0.58	0.60	0.08
	'JAPAN'	0.50	0.50	0.08
	'USA'	0.49	0.48	0.09
	'NEW ZEALAND'	0.47	0.47	0.08
	'TAIWAN'	0.34	0.34	0.00
SK	'JAPAN'	0.44	0.44	0.09
	'INDONESIA'	0.43	0.44	0.08
	'USA'	0.43	0.43	0.07
	'KIRIBATI'	0.42	0.43	0.09
	'TAIWAN'	0.42	0.42	0.07
	'KOREA'	0.42	0.42	0.09
	'MARSHALL IS'	0.42	0.42	0.08
	'NAURU'	0.42	0.44	0.08
	'PNG'	0.42	0.41	0.06
	'MALDIVES'	0.41	0.41	0.08
	'MICRONESIA'	0.41	0.41	0.08
YF	'CHINA'	0.46	0.48	0.05
	'INDONESIA'	0.43	0.44	0.10
	'USA'	0.46	0.46	0.00
	'SOLOMON ISLANDS'	0.42	0.44	0.01
	'KOREA'	0.46	0.50	0.07
	'MARSHALL IS'	0.44	0.44	0.00
	'MALDIVES'	0.44	0.43	0.09
	'MICRONESIA'	0.45	0.41	0.11
BE	'USA'	0.51	0.51	2.54e-4
	'MICRONESIA'	0.50	0.46	0.07
	'TAIWAN'	0.49	0.51	0.08
	'KOREA'	0.49	0.50	0.04
	'CHINA'	0.46	0.48	0.04
	'MALDIVES'	0.41	0.46	0.06
	'NAURU'	0.35	0.35	0.00
BT	'MALDIVES'	0.48	0.52	0.12
	'KOREA'	0.39	0.40	0.00
	'INDONESIA'	0.36	0.36	0.09
TG	'INDONESIA'	0.43	0.42	0.08



(a) AL



(b) SK

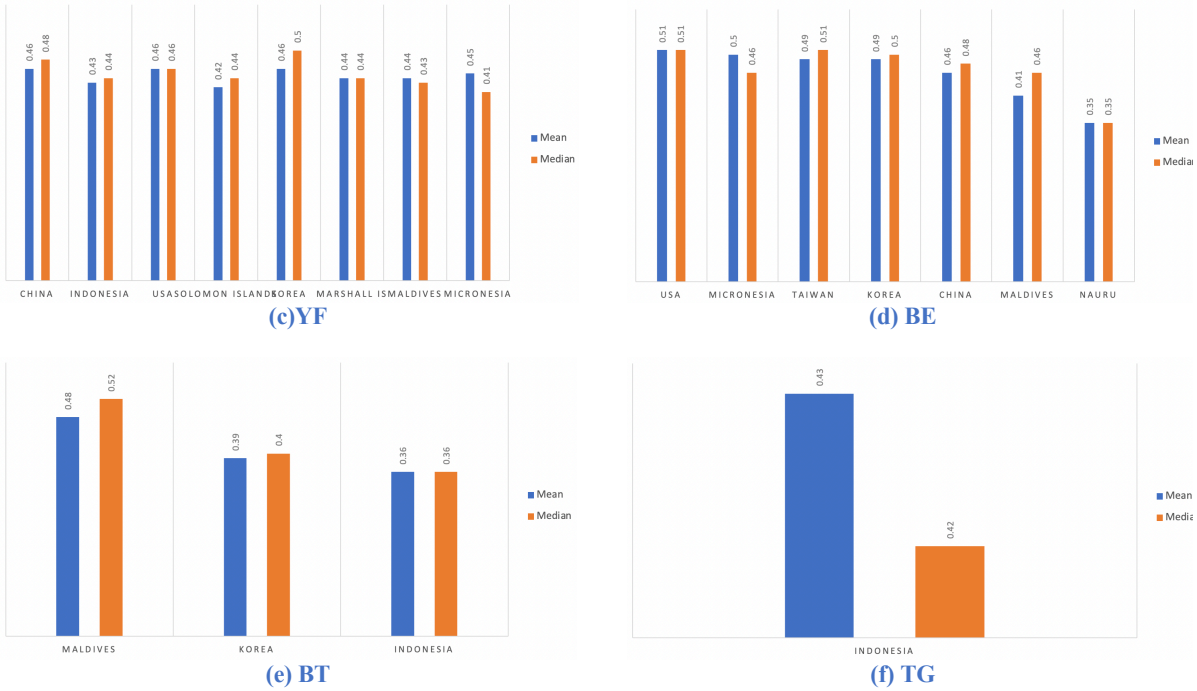


Figure 40 Summary Statistics of *PYV* for Levels of *FV_Flag* for Each Fish Species

Based on the results shown in Table 37 and Figure 40, the ordering of *PYV* across different levels of *FV-Flag* for each fish species is presented in Table 38.

Table 38 Order of *PYV* across *FV_Flag* Levels for Each Fish Species Based on Summary Statistics

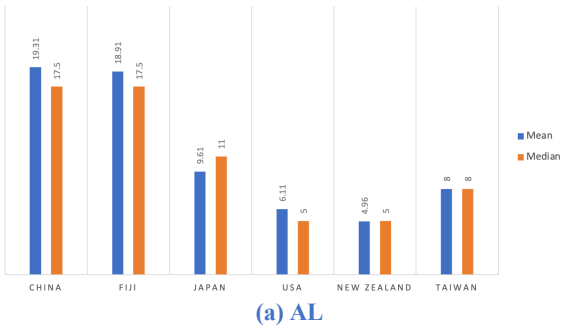
Fish Species	Order of *FV_Flag* Levels
AL	'TAIWAN' <= 'NEW ZEALAND' <= 'USA' <= 'JAPAN' <= 'CHINA' <= 'FIJI'
SK	'MICRONESIA' <= 'MALDIVES' <= 'PNG' <= 'NAURU' <= 'MARSHALL IS' <= 'KOREA' <= 'TAIWAN' <= 'KIRIBATI' <= 'USA' <= 'INDONESIA' <= 'JAPAN'
YF	'SOLOMON ISLANDS' <= 'INDONESIA' <= 'MALDIVES' <= 'MARSHALL IS' <= 'MICRONESIA' <= 'USA' <= 'KOREA' <= 'CHINA'
BE	'NAURU' <= 'MALDIVES' <= 'CHINA' <= 'KOREA' <= 'TAIWAN' <= 'MICRONESIA' <= 'USA'
BT	'INDONESIA' <= 'KOREA' <= 'MALDIVES'
TG	'INDONESIA'

5.5.3.3 Summary Statistics of *lot_size* for Each Fish Species

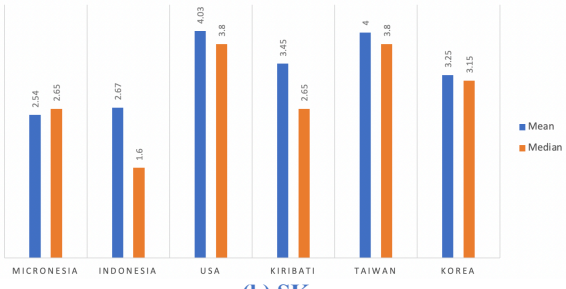
The findings have been shown in Table 39 and summarized in Figure 41.

Table 39 Summary Statistics of *lot_size* for Levels of *FV_Flag* for Each Fish Species

Fish Species	*FV_ Flags*	Mean Value	Median Value	STD
AL	'CHINA'	19.31	17.5	3.22
	'FIJI'	18.91	17.5	3.45
	'JAPAN'	9.61	11.0	2.58
	'USA'	6.11	5.0	1.97
	'NEW ZEALAND'	4.96	5.0	1.45
	'TAIWAN'	8.00	8.0	0.00
SK	'MICRONESIA'	2.54	2.65	1.76
	'INDONESIA'	2.67	1.60	1.96
	'USA'	4.03	3.80	2.02
	'KIRIBATI'	3.45	2.65	1.27
	'TAIWAN'	4.00	3.80	1.42
	'KOREA'	3.25	3.15	1.59
	'MARSHALL IS'	3.34	3.80	1.83
	'JAPAN'	6.40	6.50	0.38
	'PNG'	3.9	3.80	1.38
	'MALDIVES'	3.87	3.80	2.18
	'NAURU'	3.82	3.80	2.04
YF	'CHINA'	2.10	2.10	0.00
	'INDONESIA'	13.80	15.00	8.66
	'USA'	2.10	2.10	0.00
	'SOLOMON ISLANDS'	17.50	17.50	0.00
	'KOREA'	9.85	10.75	4.08
	'MARSHALL IS'	5.50	5.50	0.00
	'MALDIVES'	5.82	2.10	6.73
	'MICRONESIA'	9.93	9.50	5.43
BE	'USA'	4.36	2.60	2.41
	'MICRONESIA'	5.74	5.50	2.78
	'TAIWAN'	4.80	4.80	2.35
	'KOREA'	2.75	2.35	0.92
	'CHINA'	2.00	2.10	0.21
	'MALDIVES'	4.82	5.50	0.93
	'NAURU'	7.00	7.00	0.00
BT	'MALDIVES'	2.27	1.55	1.45
	'KOREA'	2.60	2.60	0.00
	'INDONESIA'	2.19	2.10	0.92
TG	'INDONESIA'	2.54	2.6	1.06



(a) AL



(b) SK

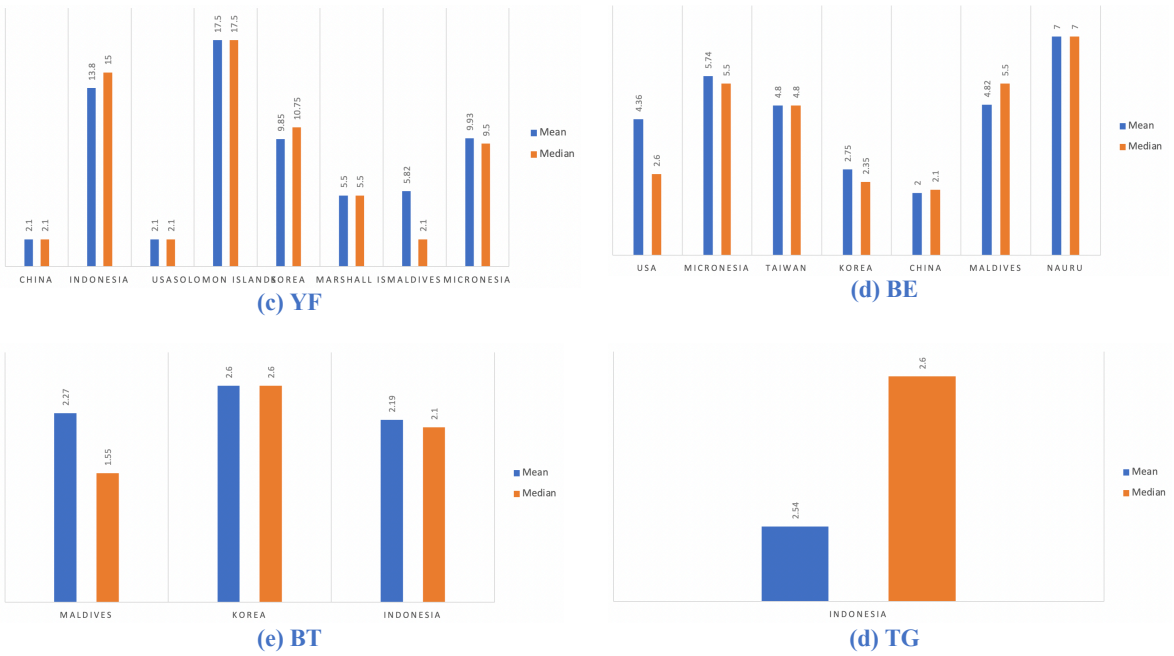


Figure 41 Summary Statistics of *Lot_ Sizes* for Levels of *FV_Flag* for Each Fish Species

Based on the results shown in Table 39 and Figure 41, the ordering of *lot_size* across different levels of *FV_Flag* for each fish species is presented in Table 40.

Table 40 Order of *lot_size* across *FV_Flag* Levels for Each Fish Species Based on Summary Statistics

Fish Species	Order of *FV-Flag* Levels
AL	'NEW ZEALAND' <= 'USA' <= 'TAIWAN' <= 'JAPAN' <= 'FIJI' <= 'CHINA'
SK	'MICRONESIA' <= 'INDONESIA' <= 'KOREA' <= 'MARSHALL IS' <= 'KIRIBATI' <= 'NAURU' <= 'MALDIVES' <= 'PNG' <= 'TAIWAN' <= 'USA' <= 'JAPAN'
YF	'CHINA' <= 'USA' <= 'MARSHALL IS' <= 'MALDIVES' <= 'KOREA' <= 'MICRONESIA' <= 'INDONESIA' <= 'SOLOMON ISLANDS'
BE	'CHINA' <= 'KOREA' <= 'USA' <= 'TAIWAN' <= 'MALDIVES' <= 'MICRONESIA' <= 'NAURU'
BT	'INDONESIA' <= 'MALDIVES' <= 'KOREA'
TG	'INDONESIA'

Based on Table 38 and Table 40, we cannot conclude that larger *lot_size* is associated with higher *PYV* across different levels of *FV_Flag* for each fish species.

5.5.3.4 Point-Biserial Correlation Coefficients

In order to study the correlation between *PYV* and different levels of *FV_Flag* for each fish species, we computed the point-biserial correlation coefficients.

Table 41 presents these findings.

Table 41 Point-Biserial Correlation Coefficients of *PYV* and *FV_Flag* for Each Fish Species

Fish Species	*FV_ Flags*	Correlation Value
AL	'CHINA'	0.23
	'FIJI'	0.27
	'JAPAN'	-0.02
	'USA'	-0.06
	'NEW ZEALAND'	-0.22
	'TAIWAN'	-0.03
SK	'MICRONESIA'	-0.02
	'INDONESIA'	0.03
	'USA'	0.03
	'KIRIBATI'	0.007
	'TAIWAN'	0.02
	'KOREA'	0.01
	'MARSHALL IS'	0.008
	'JAPAN'	0.04
	'PNG'	-0.007
	'MALDIVES'	-0.05
	'NAURU'	0.001
YF	'CHINA'	0.02
	'INDONESIA'	-0.005
	'USA'	0.007
	'SOLOMON ISLANDS'	-0.005
	'KOREA'	0.02
	'MARSHALL IS'	0.002
	'MALDIVES'	-0.003
	'MICRONESIA'	0.008
BE	'USA'	0.22
	'MICRONESIA'	0.12
	'TAIWAN'	0.08
	'KOREA'	0.07
	'CHINA'	-0.09
	'MALDIVES'	-0.28
	'NAURU'	-0.40
BT	'MALDIVES'	0.12
	'KOREA'	0.02
	'INDONESIA'	-0.11
TG	'INDONESIA'	N/A*

* TG has only one level of *FV_Flag*.

According to Table 41, the order of Point-Biserial Correlation Coefficients of *PYV* across different levels of *FV_Flag* is summarized in Figure 42.

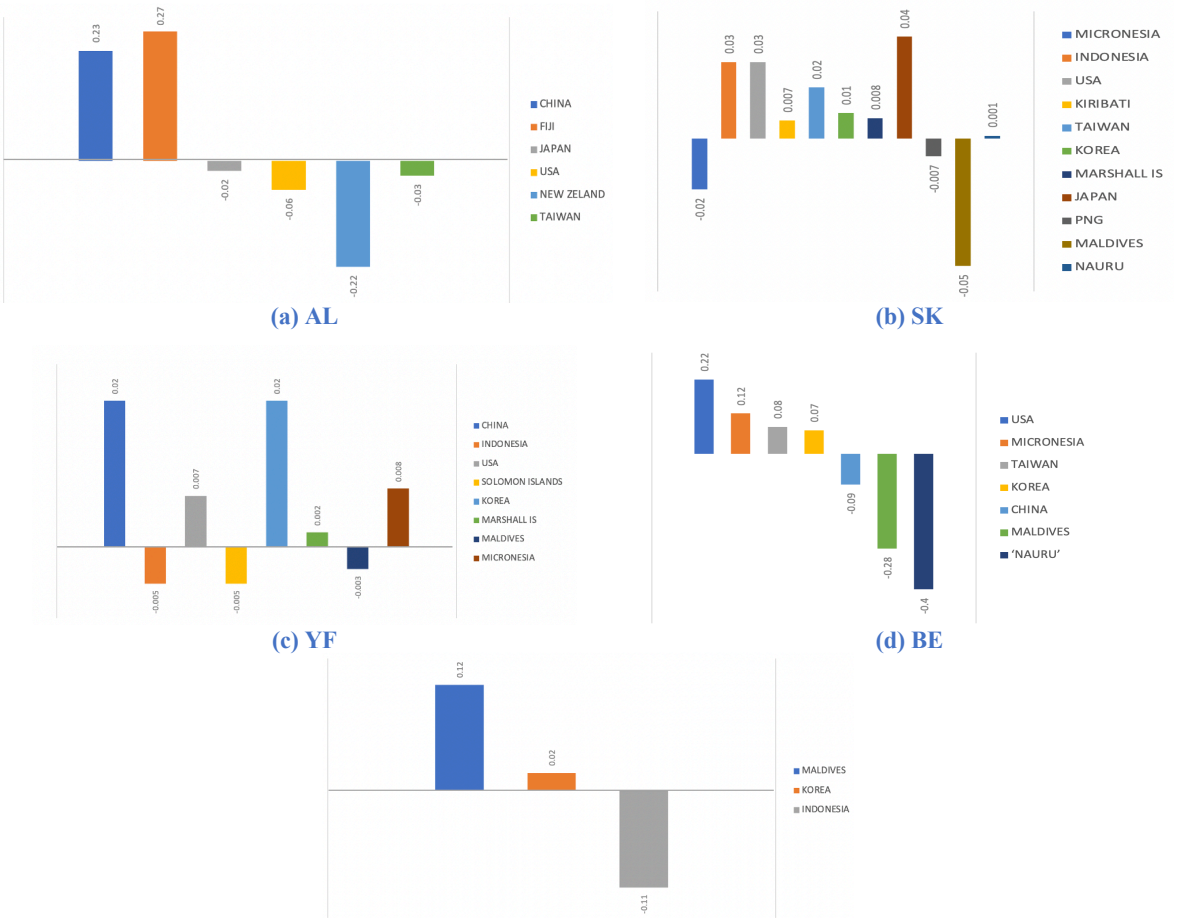


Figure 42 Point-Biserial Correlation Coefficients of *PYV* Across Levels of *FV_Flag* for Each Fish Species

5.6 Summary of EDA

In this chapter, we conducted a comprehensive analysis of quantitative process control parameters, their properties and impact on *PYV*. In Table 42, we have summarized the set of process control parameters and split them in four groups (i.e., with higher positive, higher negative, smaller positive, and smaller negative correlation with *PYV*) for all and each species.

Table 42 Summary of Process Control Parameters Impact on *PYV*

Fish Species	Parameters with Higher Positive Correlation with *PYV*	Parameters with Higher Negative Correlation with *PYV*	Parameters with Smaller Positive Correlation with *PYV*	Parameters with Smaller Negative Correlation with *PYV*
All	*Clean_bb_after_chill_temps_c_avg* *Thaw_time_min*	*Rack_fish_per_pan*	*Precook_after_spray_temps_c_avg* *Precook_racks_in_batch*	*Precook_bb_temps_c_avg*
AL	*Lot_size* *Precook_racks_in_batch* *Thaw_time_min* *Precook_after_spray_temps_c_avg* *Precook_cooking_time_min*	*Rack_internal_temps_c_avg* *Rack_total_pans* *Rack_fish_per_pan* *Rack_pans_per_rack_avg* *Precook_bb_temps_c_avg*	*Clean_bb_after_chill_temps_c_avg*	*Precook_spray_time_min*
SK	*Lot_size* *Precook_racks_in_batch* *Precook_after_spray_temps_c_avg*	*Rack_total_pans* *Rack_fish_per_pan* *Rack_pans_per_rack_avg* *Precook_spray_time_min*	*Clean_bb_after_chill_temps_c_avg* *Rack_internal_temps_c_avg* *Thaw_time_min* *Precook_cooking_time_min*	*Precook_bb_temps_c_avg*

Fish Species	Parameters with Higher Positive Correlation with *PYV*	Parameters with Higher Negative Correlation with *PYV*	Parameters with Smaller Positive Correlation with *PYV*	Parameters with Smaller Negative Correlation with *PYV*
YF	*Lot_size* *Precook_racks_in_batch* *Thaw_time_min* *Precook_after_spray_temps_c_avg* *Precook_cooking_time_min*	*Rack_total_pans* *Rack_fish_per_pan*	*Clean_bb_after_chill_temps_c_avg* *Precook_spray_time_min*	*Rack_internal_temps_c_avg* *Rack_pans_per_rack_avg* *Precook_bb_temps_c_avg*
BE	*Lot_size* *Precook_spray_time_min*	*Precook_racks_in_batch* *Rack_pans_per_rack_avg* *Thaw_time_min* *Precook_after_spray_temps_c_avg* *Precook_cooking_time_min*	*Clean_bb_after_chill_temps_c_avg* *Rack_internal_temps_c_avg* *Rack_fish_per_pan*	*Rack_total_pans* *Precook_bb_temps_c_avg*
BT	*Lot_size* *Precook_racks_in_batch* *Precook_after_spray_temps_c_avg* *Precook_cooking_time_min*	*Rack_total_pans* *Rack_pans_per_rack_avg* *Thaw_time_min*	*Clean_bb_after_chill_temps_c_avg* *Precook_spray_time_min* *Precook_bb_temps_c_avg*	*Rack_internal_temps_c_avg* *Rack_fish_per_pan*
TG	*Rack_internal_temps_c_avg* *Rack_fish_per_pan*	*Lot_size* *Precook_racks_in_batch* *Rack_pans_per_rack_avg* *Thaw_time_min* *Precook_spray_time_min* *Precook_bb_temps_c_avg* *Precook_cooking_time_min*	*Clean_bb_after_chill_temps_c_avg*	*Rack_total_pans* *Precook_after_spray_temps_c_avg*

The results of this analysis provide an important insight and serve as an input to the next phase of the DS framework that is predictive ML modeling.

Also, we analyzed qualitative categorical features from the set of raw material parameters (i.e., *CT_Method*, *CT_Area* and *FV_Flag*) for all, as well as stratified fish species to determine their optimal set generating the highest *PYV* as summarized in Table 43. These results will also be used in the next chapter.

Table 43 Recommended Optimal Set of Raw Material Parameters for Highest *PYV*

Fish Species	*CT_Method*	*CT_Area*	*FV_Flag*
All	'LONG LINE'	'EASTERN PACIFIC'	'FIJI'
AL	'LONG LINE'	'EASTERN PACIFIC'	'FIJI'
SK	'HAND LIINE'	'NORTHWEST PACIFIC'	'JAPAN'
YF	'HAND LINE'	'INDIAN OCEAN'	'CHINA'
BE	'PURSE SEINE'	'INDIAN OCEAN'	'USA'
BT	'POLE & LINE'	'INDIAN OCEAN'	'MALDIVES'
TG	'PURSE SEINE'	'WEASTERN PACIFIC'	'INDONESIA'

Chapter 6. Predictive Machine Learning Modeling of *PYV*

In this chapter, we present the application of Machine Learning (ML) algorithms for prediction of *PYV* based on the set of process control parameters and raw material parameters identified in the previous chapter. It is followed by a Stacked Modeling (SM) exercise aimed at an increased accuracy of *PYV* predictions.

6.1 Feature Engineering

Within the context of this study, feature engineering is broadly interpreted as making data better suited to the problem at hand to achieve the best results from the algorithms being applied (e.g., Kumar, 2021; Holbrook, 2021). The main activities of the feature engineering (i.e., handling the missing and imbalance values, outliers cleansing, encoding and feature selection) have been performed in two previous chapters.

In this section, we present the set of process control parameters selected for ML and SM modeling and demonstrate an encoding method applied to the raw material parameters to render them suitable for ML and SM algorithms.

6.1.1 Process Control Features

Table 44 lists process control parameters most correlated with PYV as determined in the fitted Lasso Regression analysis as shown in Figure 20 and Table 13 of Chapter 5.

Table 44 Process Control Parameters for ML

Process Control Feature	Description
Lot_size	Fish size
Thaw_time_min	Duration of fish thawing
Rack_internal_temps_c_avg	Temperature of fish taken prior to going into the precooker
Precook_cooking_time_min	Total time for cooking the fish
Precook_bb_temps_c_avg	Precooked fish backbone (BB) temperatures
Precook_spray_time_min	Spraying duration
Precook_after_spray_temps_c_avg	Fish temperature after spraying
Precook_racks_in_batch	The number of racks or trolleys that are rolled into the precooking machines
Rack_fish_per_pan	The number of fish put in a pan which is placed on the rack into precooker
Rack_total_pans	Total number of pans placed on the rack into the precooker
Rack_pans_per_rack_avg	The average number of pans placed on the racks for the precooker
Clean_bb_after_chill_temps_c_avg	Cleaned fish backbone (BB) temperatures after chilling

These parameters are chosen to form a set of process control features for ML and SM algorithms.

6.1.2 Raw Material Features

As discussed in Chapter 5, important raw material parameters are *CT_Method*, *CT_Area,* and *FV_Flag*. One of the challenges in using these parameters as features for ML and SM algorithms is their categorical type. It requires to apply a certain encoding procedure to convert the levels of categorical parameters into numeric values. Among the encoding techniques reviewed in Section 3.2.1 of Chapter 3, one-hot encoding method was applied. This is because data type of our categorical variables is nominal, as such no ordering exists for them. Table 45, Table 46, and Table 47 demonstrate the respective one-hot encoding of the three raw material parameters.

Table 45 One Hot Encoding of *CT_Method*

CT_Method
'PURSE SEINE'
'POLE & LINE'
'HAND LINE'
'LONG LINE'

'PURSE SEINE'	'POLE & LINE'	'HAND LINE'	'LONG LINE'
1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1

Table 46 One Hot Encoding of *CT_Area*

CT_Area
'WESTERN PACIFIC'
'INDIAN OCEAN'
'NORTHWEST PACIFIC'
'EASTERN PACIFIC'

'WESTERN PACIFIC'	'INDIAN OCEAN'	'NORTHWEST PACIFIC'	'EASTERN PACIFIC'
1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1

Table 47 One Hot Encoding of *FV_Flag*

FV_Flag
'JAPAN'
'TAIWAN'
'INDONESIA'
'NAURU'
'PNG'
'KOREA'
'MARSHALL IS'
'USA'

'JAPAN'	'TAIWAN'	'INDONESIA'	'NAURU'	'PNG'	'KOREA'	'MARSHALL IS'	'USA'
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0
0	0	0	1	0	0	0	0
0	0	0	0	1	0	0	0
0	0	0	0	0	1	0	0
0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	1

The encoded versions of these categorical features have been used in ML and SM algorithms for *PYV* predictive modeling.

6.2 *PYV* Prediction Using ML Algorithms

The following ML algorithms with their defined hyperparameters have been investigated for predictive modeling of *PYV*: Linear Regression with no Regularization; Support Vector Machine (SVM) with RBF kernel; Neural Network with 3 hidden layers, 100 neurons in each layer, Rectified Linear Unit (Relu) activation function, and Adam optimizer; k-Nearest Neighbour (k-NN) with k=5, uniform weight, and Euclidean metric; as well as, Random Forest with 100

Decision Trees; and AdaBoost with 100 Decision Stumps, Learning Rate of 0.001, and Linear Loss Function. The abovementioned settings of the hyperparameters have been determined to satisfy a criterion of higher computational efficiency.

The experiments have been conducted on the selected set of features from the Working dataset in both settings: all fish species and stratified fish species. The quality of fit was assessed using the most illustrative performance metrics, such as Mean-Squared Error (MSE), Root Mean-Squared Error (RMSE), Mean Absolute Error (MAE) as well as R^2.

In order to test the model's ability to predict new (unseen) data and account for problems like overfitting (Cawley et al., 2010), the k-fold cross-validation technique was applied. Different values of k in the range from 2 to 25 have been tested, and k=20 has shown the best performance. As well, different random splitting percentages of the data into training and testing sets have been verified where randomly choosing 75% of the data for training the algorithm and the rest of 25% for testing purposes has provided the best performance. These settings have been used for all modeling tasks in the study.

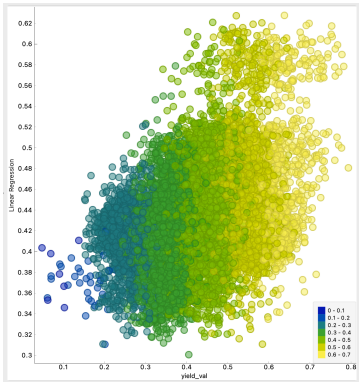
6.2.1 *PYV* Prediction for All Fish Species

The summary of predictive ML modeling for all fish species is shown in Table 48.

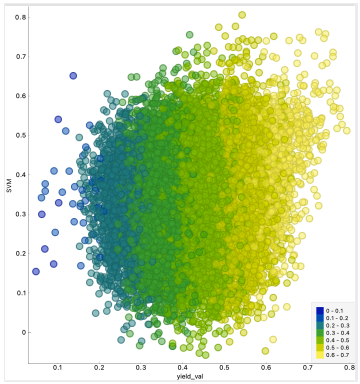
Table 48 Performance Metrics of ML Models for All Fish Species

ML Models	MSE	RMSE	MAE	R^2
Linear Regression	0.007	0.081	0.064	0.124
SVM	0.015	0.123	0.096	-0.995
Neural Network	0.008	0.088	0.066	-0.030
kNN	0.005	0.073	0.053	0.300
Random Forest	0.003	0.054	0.037	0.628
AdaBoost	0.002	0.045	0.018	0.737

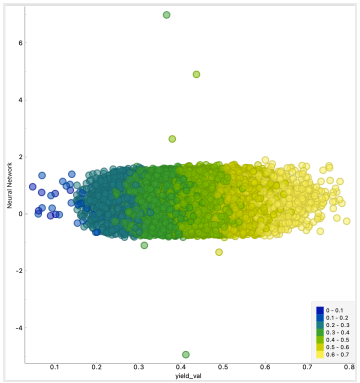
As well, the modeling results are visualized in Figure 43.



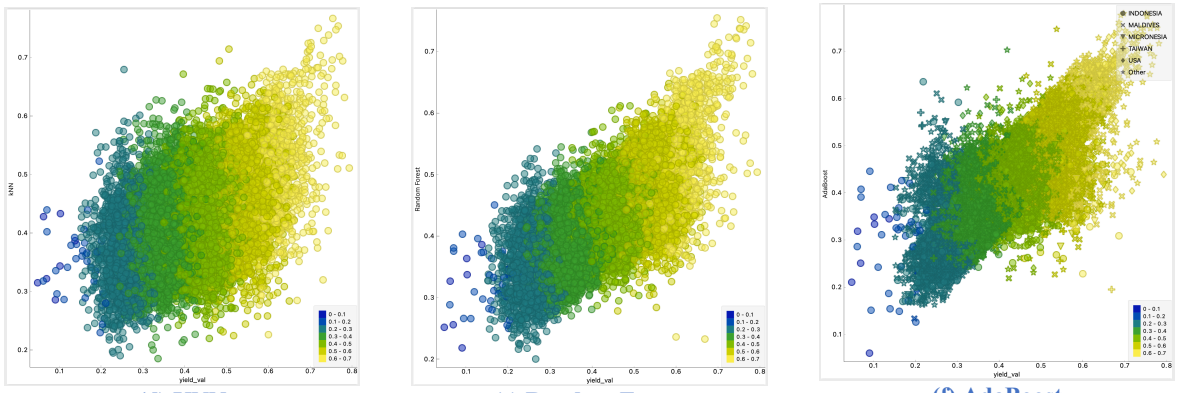
(a) Linear Regression



(b) SVM



(c) Neural Network



(d) KNN (e) Random Forest (f) AdaBoost

Figure 43 Results of *PYV* Predictive ML Modeling for All Fish Species

Additional performance comparison is shown in Figure 44.

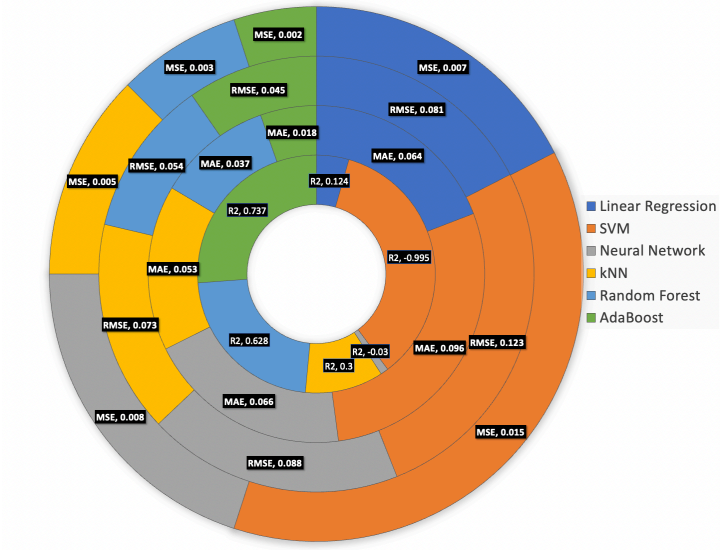


Figure 44 Performance Metrics of ML Algorithms for All Fish Species

It can be concluded from Table 48, Figure 43 and Figure 44, that the best performance in predicting of *PYV* for all fish species was achieved on Random Forest and AdaBoost models, each of which is an ensemble ML algorithm.

6.2.2 *PYV* Prediction for Each Fish Species

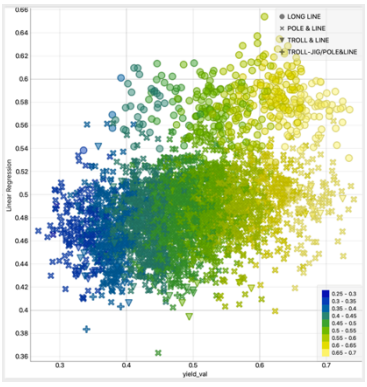
The performance of ML algorithms for each fish species is presented in Table 49.

Table 49 Performance Metrics of ML Models for Each Fish Species

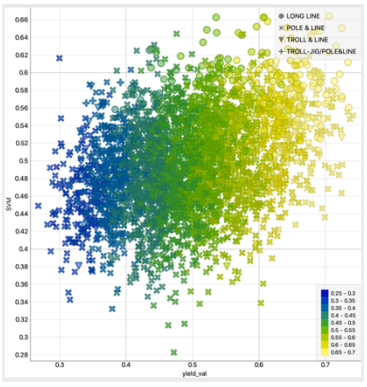
Fish Species	ML Model	MSE	RMSE	MAE	R^2
AL	Linear Regression	0.006	0.080	0.063	0.214
	SVM	0.008	0.087	0.070	0.079
	Neural Network	0.007	0.085	0.065	0.117
	kNN	0.005	0.069	0.046	0.429
	Random Forest	0.002	0.049	0.029	0.711
	AdaBoost	0.002	0.045	0.016	0.754
SK	Linear Regression	0.006	0.080	0.063	0.010
	SVM	0.016	0.126	0.101	-1.455
	Neural Network	0.007	0.083	0.065	-0.064
	kNN	0.005	0.070	0.052	0.235
	Random Forest	0.003	0.050	0.034	0.611
	AdaBoost	0.002	0.042	0.017	0.723
YF	Linear Regression	0.007	0.085	0.066	0.071
	SVM	0.012	0.107	0.086	-0.478
	Neural Network	0.008	0.088	0.067	0.020
	kNN	0.006	0.075	0.054	0.284
	Random Forest	0.003	0.058	0.038	0.572
	AdaBoost	0.003	0.055	0.023	0.612
BE	Linear Regression	0.004	0.061	0.049	0.490
	SVM	0.006	0.078	0.071	0.169
	Neural Network	0.027	0.163	0.107	-2.656
	kNN	0.005	0.069	0.050	0.341
	Random Forest	0.002	0.048	0.036	0.687
	AdaBoost	0.002	0.046	0.021	0.712
BT	Linear Regression	0.008	0.090	0.068	0.021
	SVM	0.009	0.097	0.078	-0.155
	Neural Network	0.011	0.103	0.079	-0.282
	kNN	0.007	0.087	0.065	0.086
	Random Forest	0.005	0.071	0.049	0.394
	AdaBoost	0.004	0.066	0.034	0.469
TG	Linear Regression	0.007	0.081	0.065	0.000
	SVM	0.006	0.077	0.065	0.112
	Neural Network	0.013	0.115	0.089	-0.997
	kNN	0.005	0.073	0.056	0.198
	Random Forest	0.003	0.052	0.036	0.599
	AdaBoost	0.002	0.043	0.019	0.715

As well, the outcome of *PYV* predictive modeling is visualized in Figure 45 to Figure 50.

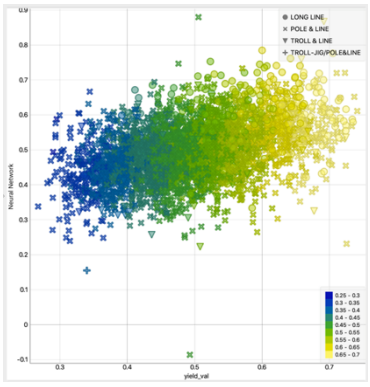
For AL



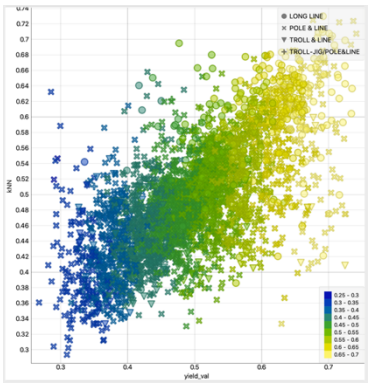
(a) Linear Regression



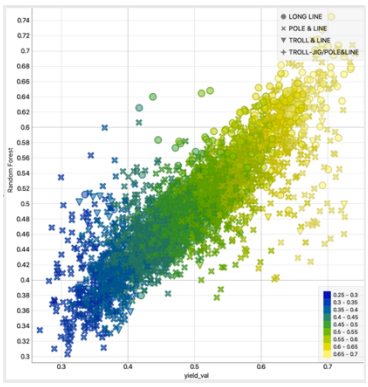
(b) SVM



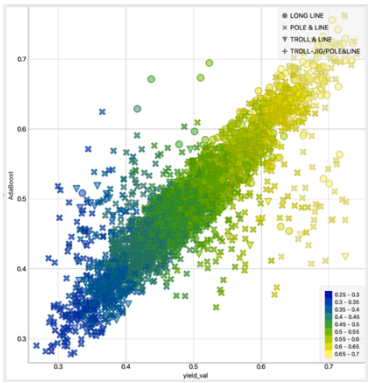
(c) Neural Network



(d) KNN



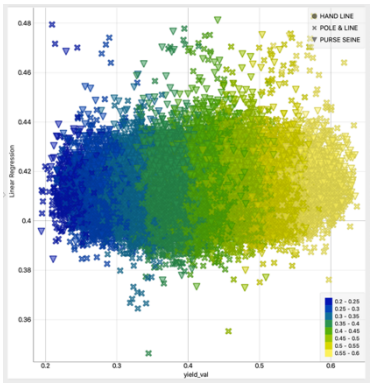
(e) Random Forest



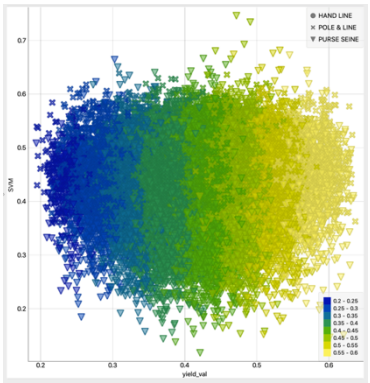
(f) AdaBoost

Figure 45 Results of *PYV* Predictive ML Modeling for AL

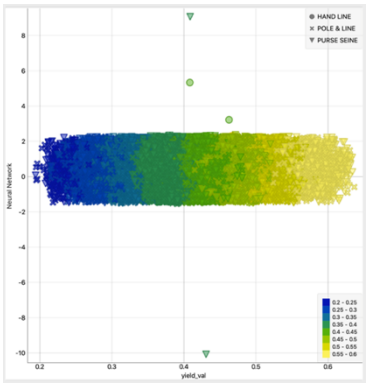
For SK



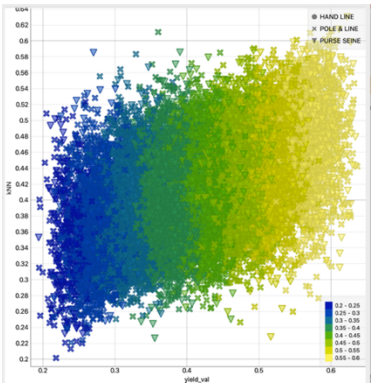
(a) Linear Regression



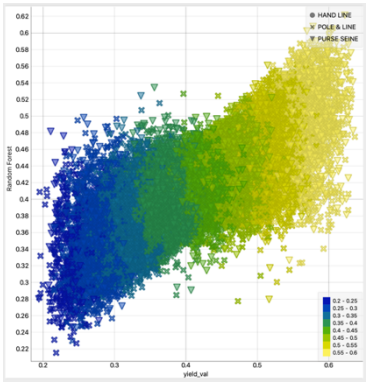
(b) SVM



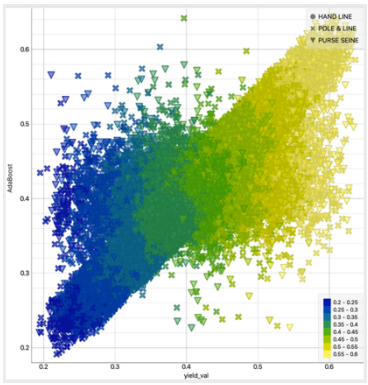
(c) Neural Network



(d) KNN



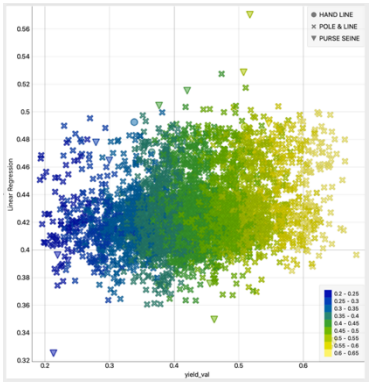
(e)Random Forest



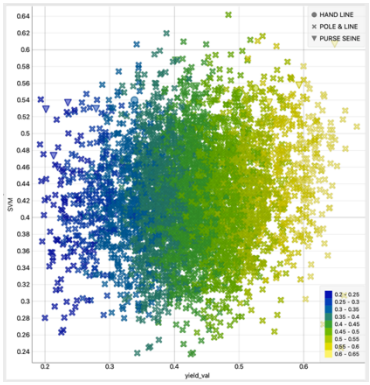
(f)AdaBoost

Figure 46 Results of *PYV* Predictive ML Modeling for SK

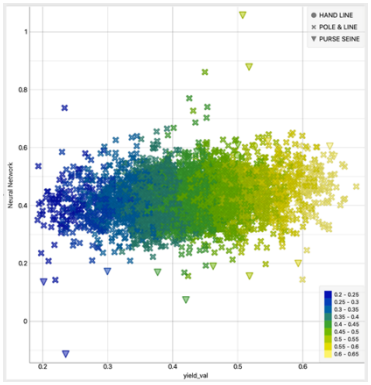
For YF



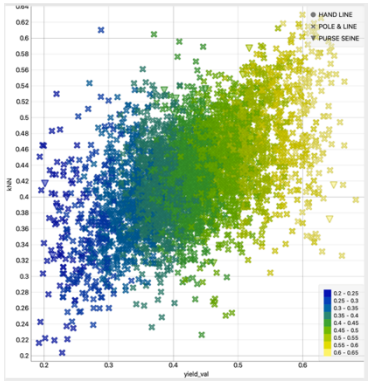
(a) Linear Regression



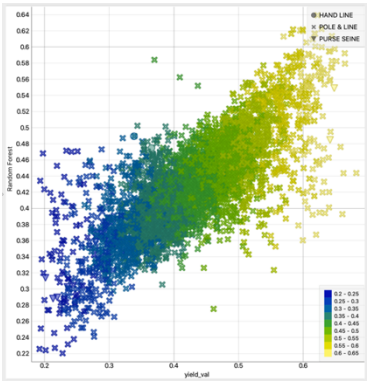
(b) SVM



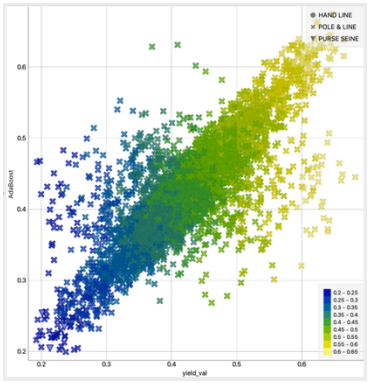
(c) Neural Network



(d) KNN



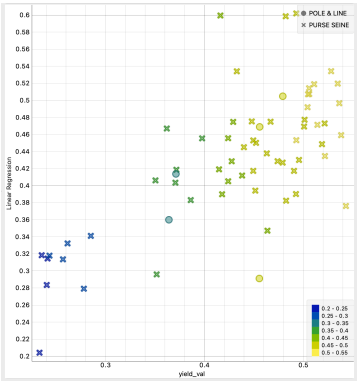
(e) Random Forest



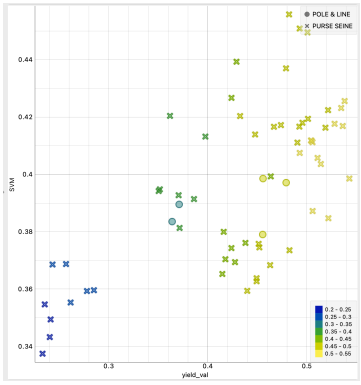
(f) AdaBoost

Figure 47 Results of *PYV* Predictive ML Modeling for YF

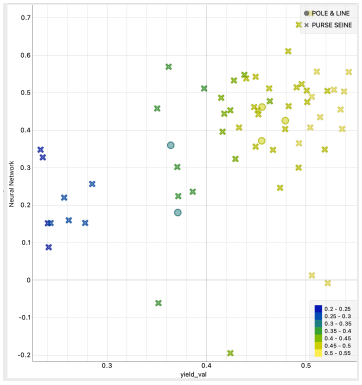
For BE



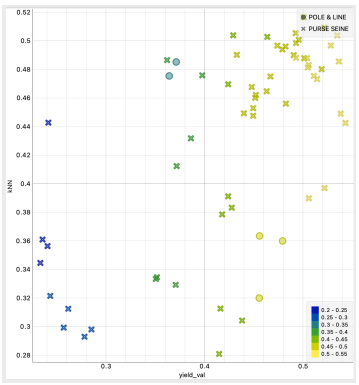
(a) Linear Regression



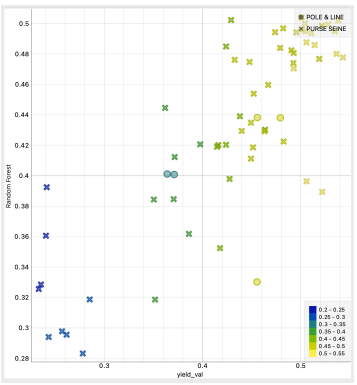
(b) SVM



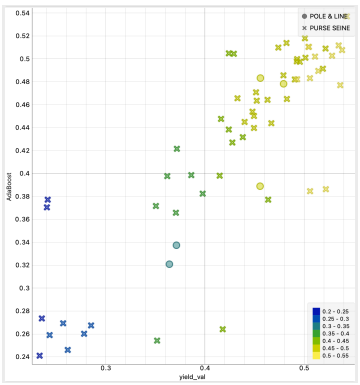
(c) Neural Network



(d) KNN



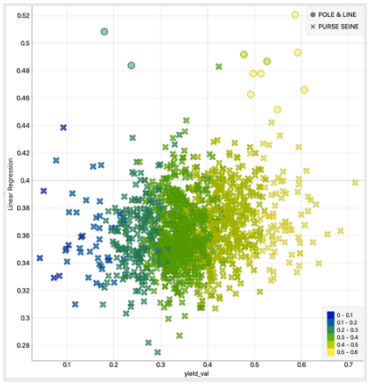
(e) Random Forest



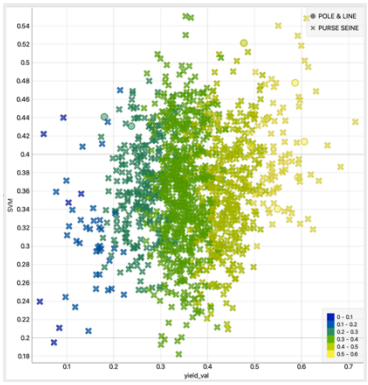
(f) AdaBoost

Figure 48 Results of *PYV* Predictive ML Modeling for BE

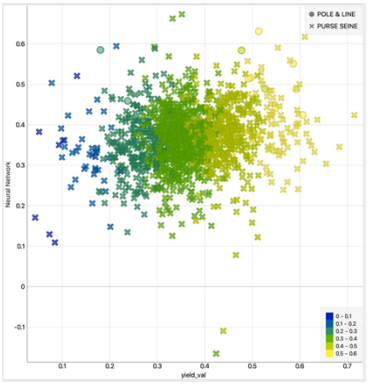
For BT



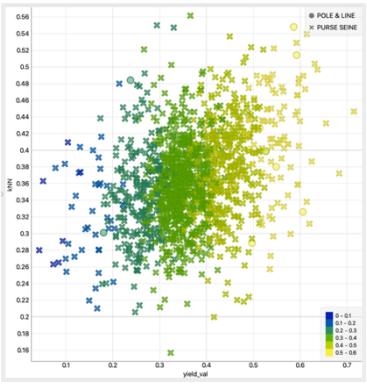
(a) Linear Regression



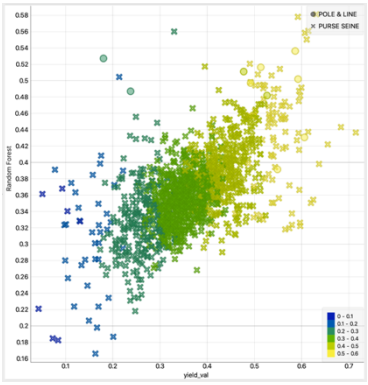
(b) SVM



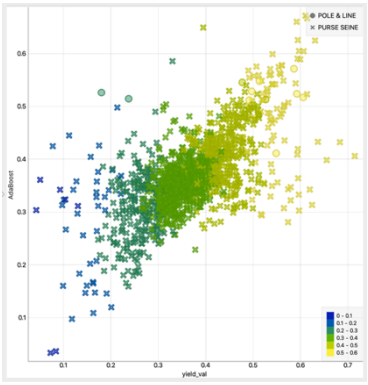
(c) Neural Network



(d) KNN



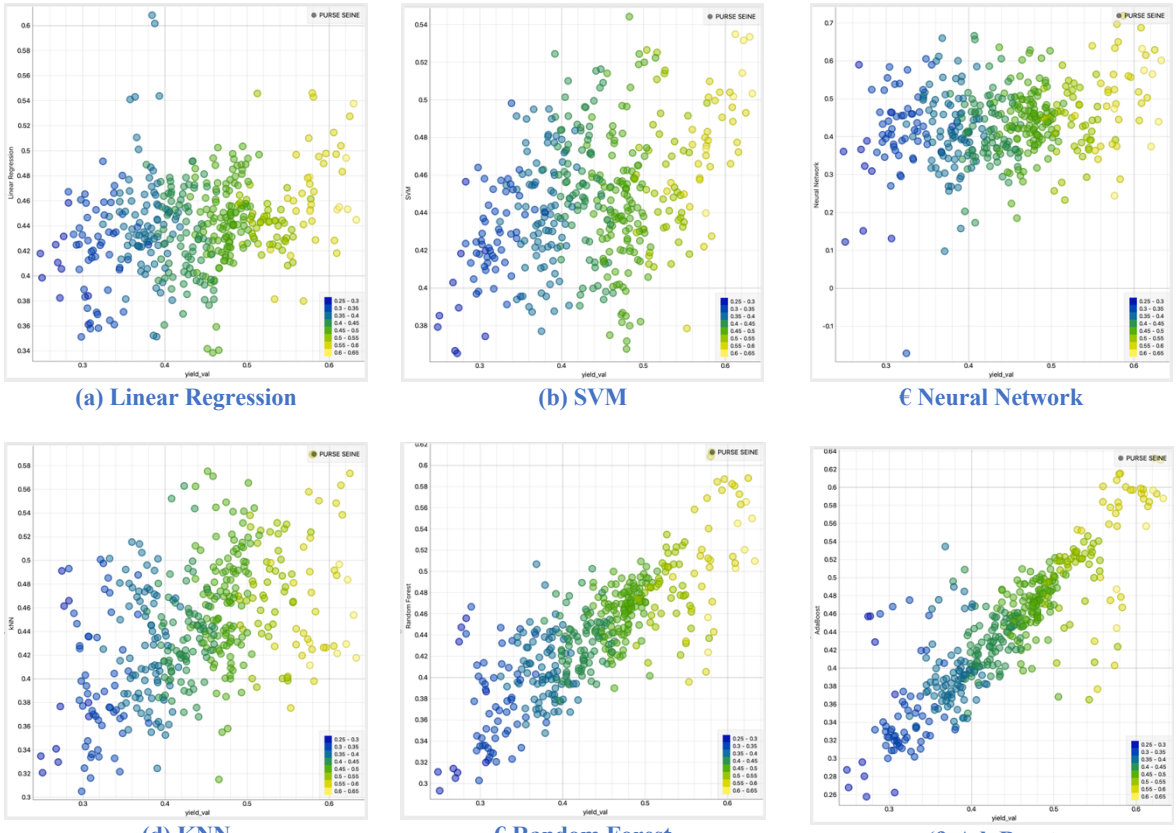
(e) Random Forest



(f) AdaBoost

Figure 49 Results of *PYV* Predictive ML Modeling for BT

For TG

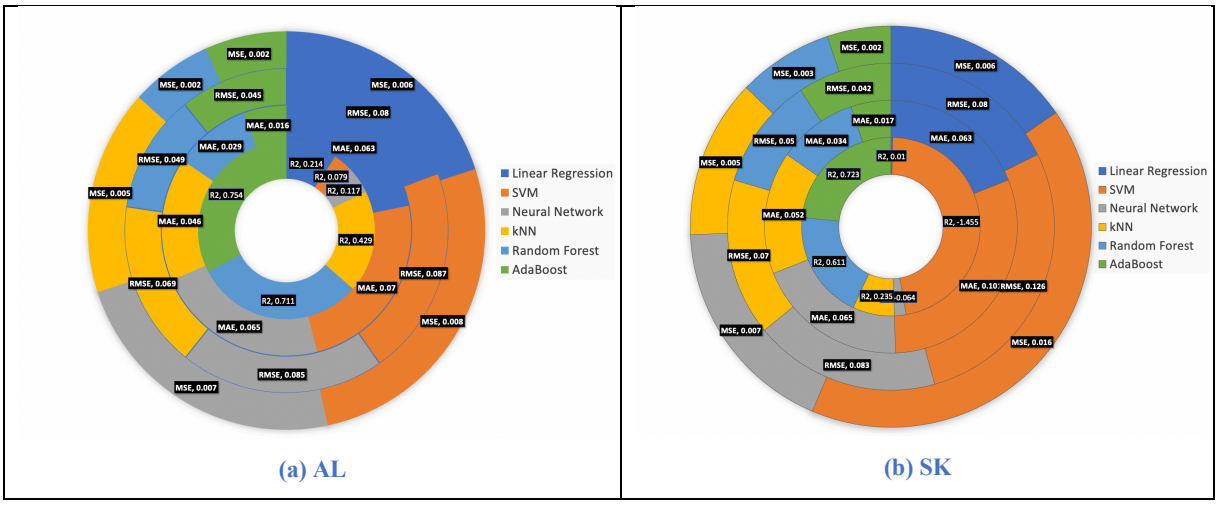


(a) Linear Regression (b) SVM € Neural Network

(d) KNN € Random Forest (f) AdaBoost

Figure 50 Results of *PYV* Predictive ML Modeling for TG

Additional performance comparison is shown in Figure 51.



(a) AL (b) SK

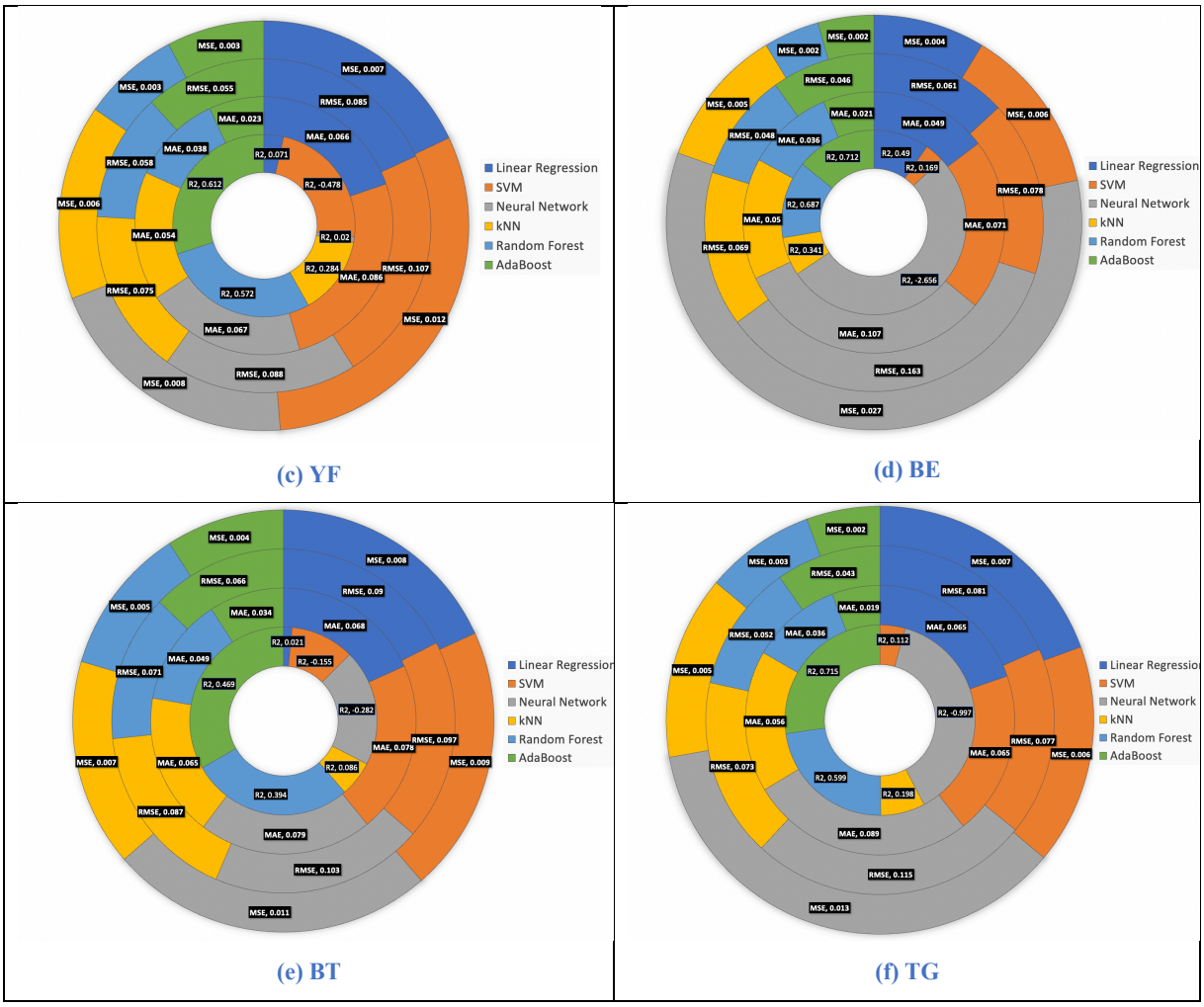


Figure 51 Performance Metrics of ML Algorithms for Each Fish Species

Same as in the case of all fish species, it can be concluded from Table 48 and Figure 43 that the best performance in predicting of *PYV* for each fish species was achieved on Random Forest and AdaBoost models, each of which is an ensemble ML algorithm.

6.3 *PYV* Prediction Using Stacked Modeling

As shown in previous sections, ensemble ML algorithms (namely, Random Forest and AdaBoost), provided the best results in terms of *PYV* predictions in both settings: for all fish species and stratified fish species. We conducted an exploratory study of the Stacked Modeling (SM) approach to test a possible performance improvement of the *PYV* predictions. Stacking is a meta-learning algorithm combining predictions from two or more base ML models for improved accuracy (Himmetoglu, 2017). The SM architecture applied in this research is shown in Figure 52.

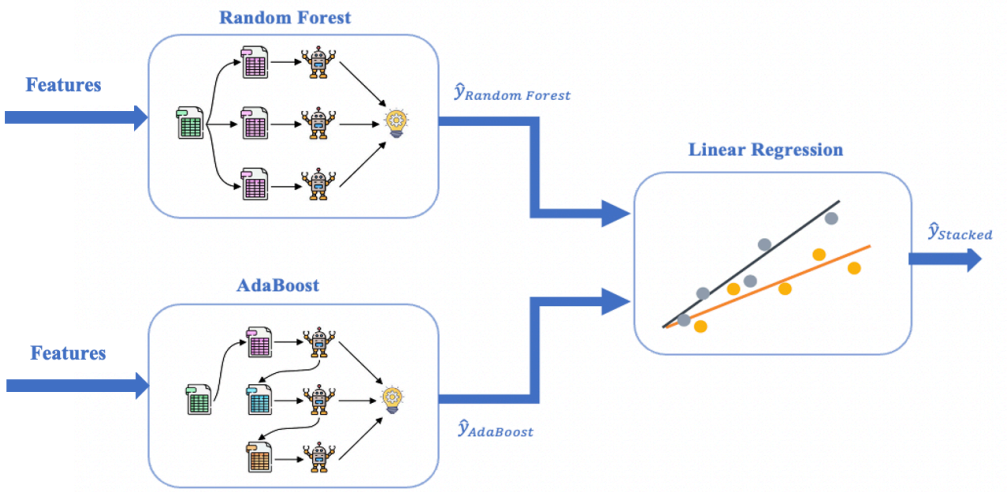


Figure 52 Stacked Modeling Architecture

6.3.1 All Fish Species

The results of SM for all fish species are reported in Table 50 and Figure 53.

Table 50 Performance Metrics of SM Models for All Fish Species

Prediction Model	MSE	RMSE	MAE	R^2
Stacked Model	0.002	0.044	0.015	0.756

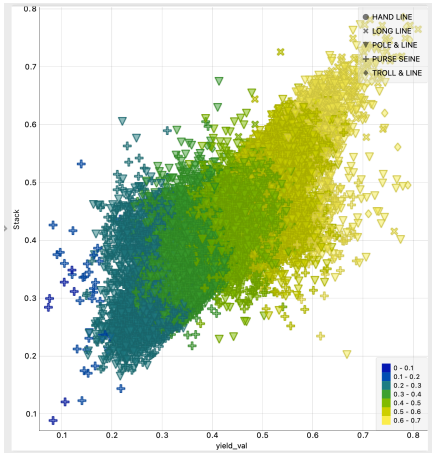


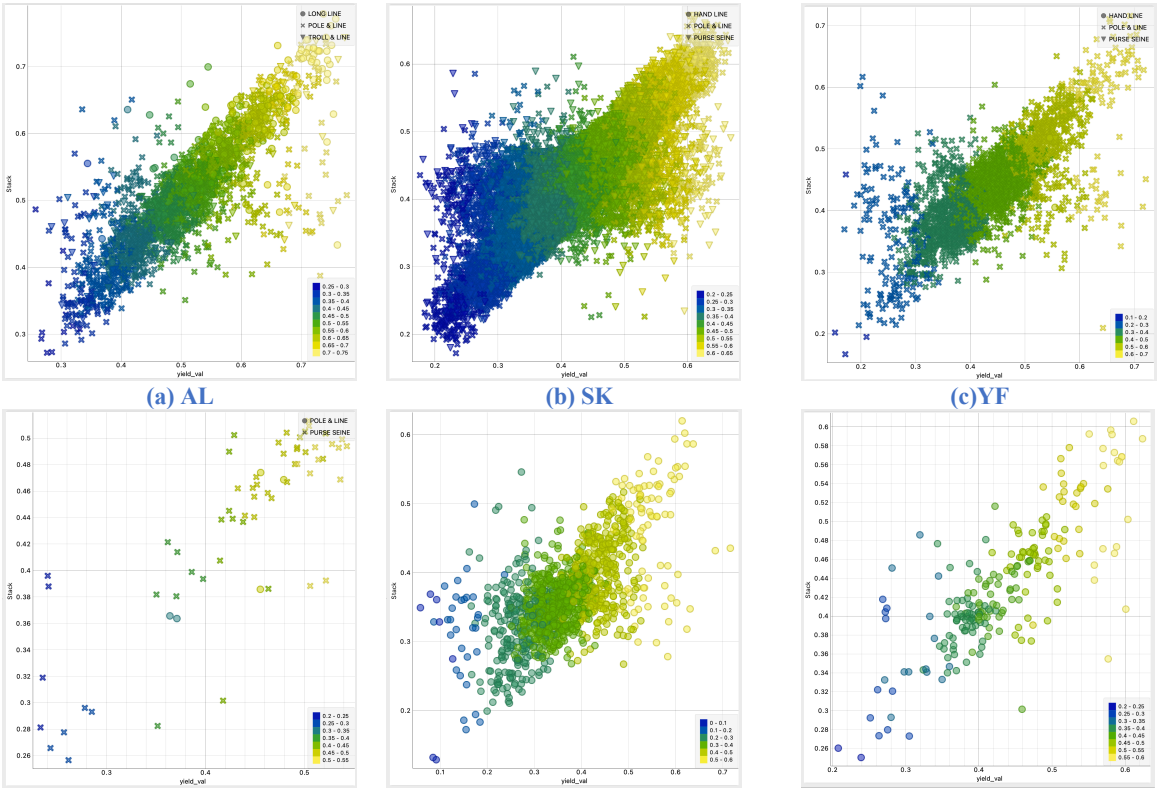
Figure 53 SM Results of *PYV* Prediction for All Fish Species

6.3.2 For Each Fish Species

The results of SM for all fish species are reported in Table 51 and Figure 54.

Table 51 Performance Metrics of SM Models for Each Fish Species

Fish Species	MSE	RMSE	MAE	R^2
AL	0.002	0.045	0.015	0.757
SK	0.002	0.042	0.016	0.725
YF	0.003	0.054	0.022	0.621
BE	0.002	0.046	0.020	0.714
BT	0.004	0.066	0.033	0.471
TG	0.002	0.043	0.018	0.717



(a) AL (b) SK (c)YF

(d) BE (e) BT (f) TG

Figure 54 SM Results of PYV Prediction for Each Fish Species

6.4 Summary of ML and SM Predictive Modeling

As noted above, Random Forest and AdaBoost ML algorithms provided the best learning performance for prediction of *PYV*. On the other hand, SM architecture did not significantly improve the prediction accuracy compared to standalone ML ensemble models (see Table 52). However, sometimes even small improvements may generate meaningful economic beneficial contributions (Himmetoglu, 2017).

In particular, even an average improvement of 0.55% in terms of R^2 of *PYV* prediction performance, will result in substantially lesser waste in production, which directly could save hundreds of thousands or even millions of dollars a year depending on the scale of the seafood processor (Tamm, 2020). Obviously, this is a significant impact on economic efficiency of seafood production.

Table 52 Performance Metrics Summary of Random Forest, AdaBoost, and SM

Fish Species	ML Modeling	MSE	RMSE	MAE	R^2
All	Random Forest	0.003	0.054	0.037	0.628
	AdaBoost	0.002	0.045	0.018	0.737
	Stacking	0.002	0.044	0.015	0.756
AL	Random Forest	0.002	0.049	0.029	0.711
	AdaBoost	0.002	0.045	0.016	0.754
	Stacking	0.002	0.045	0.015	0.757
SK	Random Forest	0.003	0.050	0.034	0.611
	AdaBoost	0.002	0.042	0.017	0.723
	Stacking	0.002	0.042	0.016	0.725
YF	Random Forest	0.003	0.058	0.038	0.572
	AdaBoost	0.003	0.055	0.023	0.612
	Stacking	0.003	0.054	0.022	0.621
BE	Random Forest	0.002	0.048	0.036	0.687
	AdaBoost	0.002	0.046	0.021	0.712
	Stacking	0.002	0.046	0.020	0.714
BT	Random Forest	0.005	0.071	0.049	0.394
	AdaBoost	0.004	0.066	0.034	0.469
	Stacking	0.004	0.066	0.033	0.471
TG	Random Forest	0.003	0.052	0.036	0.599
	AdaBoost	0.002	0.043	0.019	0.715
	Stacking	0.002	0.043	0.018	0.717

The investigated modeling strategies on the basis of ensemble methods of AdaBoost (as a boosting-based algorithm), Random Forest (as a Bagging-based algorithm) and stacked architecture of the two of them with a linear regression for predicting the final output enables us to reliably predict *PYV* for various settings of the raw material and process control parameters. This will be economically beneficial for seafood production facilities as they can perform their business planning and forecasting in a more informed and predictive way. Given that *PYV* is a complementary indicator to the waste rate, prediction of *PYV* and its maximization would mean at the same time reduction of the waste rate and thus contribute not only to the economic well-being of the individual seafood producers but also to the overall sustainability of the seafood industry.

Chapter 7. Discussion and Conclusions

This research contributes to the body of knowledge in the field of DS and ML by applying corresponding tools and techniques to the domain of seafood production. To the best of our knowledge, there aren't prior comparable studies reported in the literature as detailed electronic datasets covering each technological stage, spanning from catching the raw Tuna fish all the way through to the final production of Tuna cans, had never been available to the researchers. ThisFish Inc. is the first company which invented a unique Tally software system and successfully collected electronic real-world datasets in a seafood production facility located in Bangkok, Thailand, over a 2-year period from January 2018 until November 2020. Therefore, the novelty of this research has been in conducting a comprehensive analytical and modeling studies on these datasets following a common methodological basis.

In this thesis, a Data Science (DS) approach as a methodological foundation has been applied to a broad spectrum of theoretical and practical issues in the seafood production industry. On the basis of this approach, we suggested and elaborated a DS framework, phases in its development lifecycle and a common roadmap to be followed in the future projects in the domain. This addresses RQ1 and the details on the DS framework are given in Chapter 3.

On the first step of the proposed roadmap, Collection, Extraction, Transformation and Loading (CETL) tasks were performed including datasets extraction, data stratification and mapping, data cleansing (e.g., outlier removal, null values treatment) and redundancy elimination. Also, notion of seafood production yield value was formalized and mathematical articulation for its calculation was formulated based on specific circumstances of data collection by the operator. This addresses RQ2 and the corresponding results are presented in Chapter 4.

In the next step, in Chapter 5, normality analysis and normalization of the datasets took place followed by investigation of the impact of various Process Control parameters (which are numeric features) on *PYV* using statistical tools of Lasso regression coefficients and Pearson correlation coefficients for all and each individual fish species. This step addresses RQ3 and the details of this study are presented in Section 5.3, as well, the conclusions are summarized in Table 42. Separately, we studied the impact of various Raw Material parameters (which are categorical features) on *PYV* using such statistical techniques as ANOVA, independent *t*-test and point-biserial correlation coefficients. As the outcome of this step, we identified the set of numeric as well as categorical features with the highest impact on *PYV*. This step addresses RQ4 and the details of

this latter analysis, for all and each individual fish species, are provided in Sections 5.4 and 5.5, respectively. Table 43 summarizes the results.

These abovementioned findings have been utilized for performing feature engineering and extraction tasks on the Process Control and Raw Material parameters where the encoding of categorical variables has also been applied. This investigation addresses RQ5.

On the modeling step, in Chapter 6, addressing RQ6, six common ML algorithms have been investigated aiming to design a predictive ML modeling component of the framework. These methods are drawn from Linear, Deep Learning, Kernel-based, Non-Parametric, Bagging, and Boosting approaches. On the resultant datasets, these six ML methods have been tested and compared in the study: (1) Linear Regression; (2) Support Vector Machine (SVM); (3) Neural Network; (4) k-Nearest Neighbour (k-NN); (5) Random Forest; and (6) AdaBoost, using k-fold cross-validation technique with k=20 for training and testing with 75%-25% splitting percentages. The performance metrics of these methods have been evaluated and provided in Table 48 and Table 49.

It was concluded that the best performance metrics (i.e., MSE, RMSE, MAE, and R^2) have been achieved on Random Forest and AdaBoost ML algorithms, which addresses RQ7. The ability of a Stacked Modeling (SM) architecture to improve the predictive power of the ML models was studied and demonstrated a better modeling performance, which addresses RQ8. The results are summarized in Table 52.

To summarize, the top seven findings of this research are:

1. A novel Data Science framework for comprehensive analysis and predictive modeling in the seafood production industry is suggested and elaborated including its structural components, phases in the development lifecycle, and a common roadmap for future projects in the domain.
2. The notion of seafood Production Yield Value (*PYV*) was formalized as well as its mathematical articulation and programming implementation are proposed.
3. Process Control parameters in the seafood production have been investigated using statistical tools, such as Lasso regression and Pearson correlation, and four their groups determined (i.e., with higher positive, higher negative, smaller positive, and smaller negative correlation with *PYV*), to inform the industry producers.

4. Raw Material parameters in the seafood production have been investigated to determine their optimal set generating the highest *PYV* using statistical techniques, such as ANOVA, One-tailed independent *t*-test, Point-biserial correlation, Lasso regression, and summary statistics to be factored in the procurement process.
5. Various techniques for categorical features encoding have been investigated and one-hot encoding method recommended as the best suitable approach for the seafood dataset, also with a view of using in the ML algorithms.
6. Six common ML algorithms have been investigated aiming to design a predictive ML modeling component of the Data Science framework and found that the best performance metrics in predicting *PYV* are delivered by Random Forest and AdaBoost methods.
7. Study of the Stacked Modeling architecture has been carried out and its ability to improve the predictive capabilities of ML algorithms demonstrated.

The optimization module of the DS framework is aimed at maximizing *PYV* and, inversely, minimizing the waste rates through searching for optimal values of Process Control and Raw Material parameters based on the best derived predictive ML model for *PYV*. Suitable optimization methods may include Exhaustive search (Abdelkader et al., 2020), Gradient descent (Ruder, 2017), and Genetic algorithms (Rangel-Merino et al., 2005; Sivanandam et al., 2008). A detailed investigation of these techniques and shaping the optimization module is one of the future directions in this study.

Implementation of the entire framework and its deployment as a software tool in seafood production facilities is the final phase of the roadmap which can be done via dashboard interface (see Figure 55).

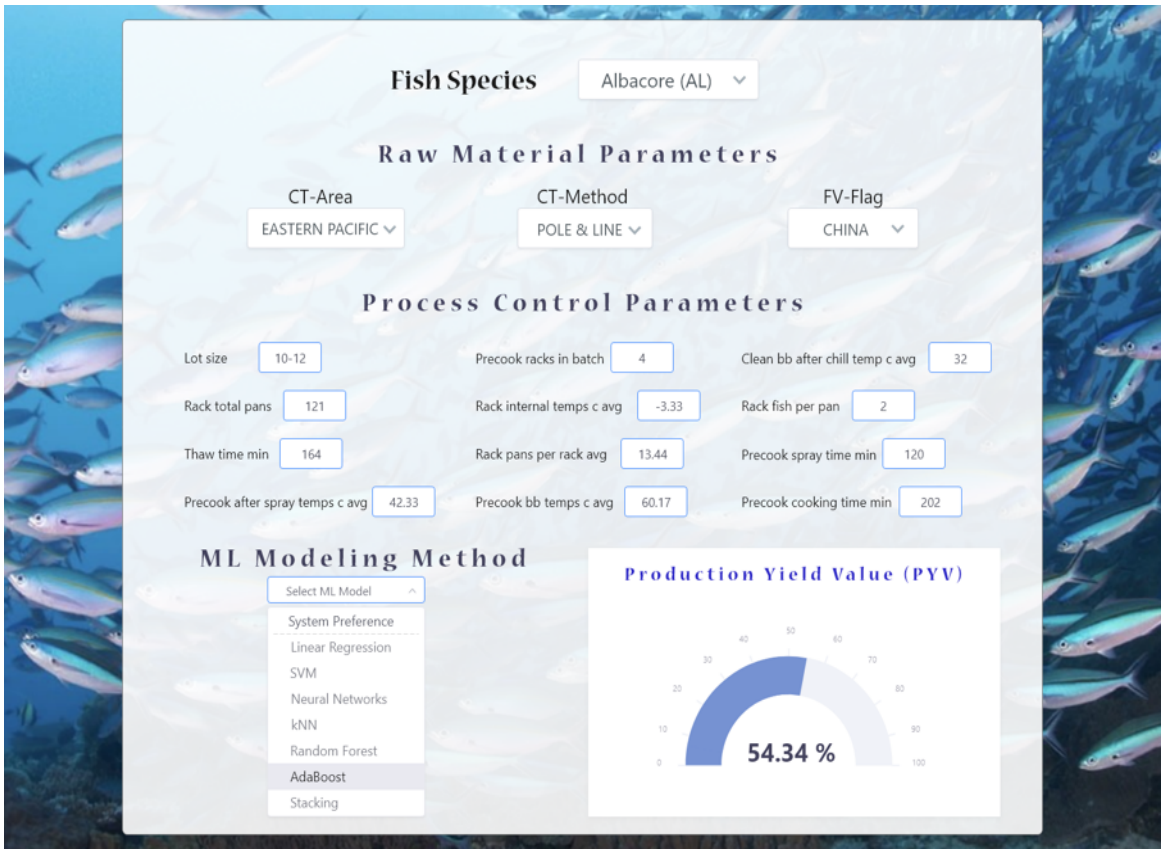


Figure 55 A Dashboard Interface

This tool can be used by executive managers in the facility to predict *PYV* based on their input of the values of Raw Material and Process Control parameters. An important practical task is to set the optimal values of the Process Control parameters which would maximize *PYV* on the given characteristics of the raw materials. Upon choosing the fish species of interest and entering the values of the Raw Material parameters (i.e., *CT_Method*, *CT_Area*, and *FV_Flag*), the system will suggest the best set of Process Control parameters maximizing *PYV*. There is an option to apply either a default ML model ("system preference") or the one of user's choice from the provided list of models.

It is anticipated that seafood producers will benefit from using the tool because even small percentage of *PYV* improvement would save them hundreds of thousands or even millions of dollars a year (Himmetoglu, 2017; Tamm, 2020).

Bibliography

Abdelkader, E., Marzouk, M., & Zayed, T. (2020). A self-adaptive exhaustive search optimization-based method for restoration of bridge defects images . *International Journal of Machine Learning and Cybernetics*, 1659–1716 .

Adams, F., Nolte, F., Colton, J., De Beer, J., & Weddig, L. (2018). Precooking as a Control for Histamine Formation during the Processing of Tuna: An Industrial Process Validation. *Journal of Food Protection, 81*(3), 444-455.

Administration, U. F. (2017). *Fish and fishery products hazards and controls guide.* Retrieved from http://www.fda.gov/ downloads/Food/GuidanceRegulation/UCM251970.pdf

Agrawal, A., & Choudhary, A. (2016). Perspective: Materials informatics and big data Realization of the "fourth paradigm" of science in materials science. *APL Mater, 4*(5).

Aha, D. W., & Kibler, D. (1991). Instance-based learning algorithms. *Machine Learning, 6*, 37-66.

Alkharusi, H. (2012). Categorical Variables in Regression Analysis: A Comparison of Dummy and Effect Coding. *International Journal of Education, 4*(2), 202-210.

Allen, M. P. (1997). *Understanding regression analysis.* New York: Plenum Press.

Alpaydin, E. (2020). *Introduction to Machine Learning (Fourth ed.).* Cambridge: MIT Press.

Al-Sahaf, H., Bi, Y., Chen, Q., Lensen, A., Mei, Y., Sun, Y., . . . Zhang, M. (2019, July). A survey on evolutionary machine learning. *Journal of the Royal Society of New Zealand*, 205-228. Retrieved from www.sas.com/en_us/insights/analytics/machine-learning.html

Altman, N. S. (1992). An introduction to kernel and nearest-neighbour nonparameteric regression. *The American Statistician, 46*(3), 175-185.

Altschul, S., Madden, T., Schäffer, A., Zhang, J., Zhang, Z., Miller, W., & Lipman, D. (1997). Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Research, 25*(17), 3389–3402.

Appiahene, P., Missah, Y. M., & Najim, U. (2020). Predicting Bank Operational Efficiency Using Machine Learning Algorithm: Comparative Study of Decision Tree, Random Forest, and Neural Networks. *Advances in Fuzzy Systems, 2020*, 1-12.

Arvanitoyannis, I. S., & Kassaveti, A. (2008). Fish industry waste: treatments, environmental impacts, current and potential uses. *International Journal of Food Science and Technology, 43*(4), 726-745.

Assarzadeh, S., & Ghoreishi, M. (2008). Neural-network-based modeling and optimization of the electro-discharge machining process. *International Journal of Advanced Manufacturing Technology, 39*(5), 488-500.

Ballestar, M., Grau-Carles, P., & Sainz, J. (2018). Predicting customer quality in e-commerce social networks: a machine learning approach. *Review of Managerial Science, 15*(5).

Bell, J. W., Farkas, B., Hale, S. A., & Lanier, T. (2001). Effect of Thermal Treatment on Moisture Transport during Steam Cooking of Skipjack Tuna (Katsuwonas pelamis). *Journal of Food Science, 66*, 307-313.

Ben-Hur, A., Horn, D., Siegelmann, H., & Vapnik, V. N. (2001). Support-Vector Clustering. *Journal of Machine Learning Research, 2*, 125-137.

Biessmann, F., Golebiowski, J., Rukat, T., Lange, D., & Schmidt, P. (2021). Automated Data Validation in Machine Learning Systems. *IEEE Computer Society Technical Committee on Data Engineering, 18*, 51-65.

Bishop, C. (1995). *Neural Networks for Pattern Recognition.* Oxford University Press.

Bouckaert, R. R. (2004). Naive Bayes classifiers that perform well with continuous variables. *AI 2004: Advances in Artificial Intelligence*, 1089–1094.

Breiman, L. (1996). Stacked regressions. *Machin Learning , 24*, 49-64.

Breiman, L. (2001). Random forests. *Machine Learning, 45*, 5–32.

Burbidge, R., Trotter, M., Buxton, B., & Holden, S. (2001). Drug design by machine learning: support vector machines for pharmaceutical data analysis. *Computers and Chemistry, 26*, 5–14.

Butler, K. T., Davies, D. W., Cartwright, H., Isayev, O., & Walsh, A. (2018). Machine learning for molecular and materials science. *Nature, 559*, 547–555.

Carbonneau, R., Laframboise, K., & Vahidov, R. (2008). Application of machine learning techniques for supply chain demand forecasting. *European Journal of Operational Research, 184*(3), 1140-1154.

Carey, G. (2017). Coding Categorical Variables. psych.colorado.edu/.

Carroll, J., Koukoura, S., McDonald, A., Charalambous, A., Weiss, S., & McArthur, S. (2018). Wind turbine gearbox failure and remaining useful life prediction using machine learning techniques. *Wind Energy, 22*, 360-375.

Carvalho, T. P., Soares, F. A., Vita, R., Francisco, R. P., Basto, J. P., & Alcalá, S. G. (2019). A systematic literature review of machine learning methods applied to predictive maintenance. *Computers and Industrial Engineering, 137*, 106024.

Cavalcante, R. C., Brasileiro, R. C., Souza, V. L., Nobrega, J. P., & Oliveira, A. L. (2016). Computational Intelligence and Financial Markets: A Survey and Future Directions. *Expert Systems With Applications, 55*, 194-211.

Cawley, G. C., & Talbot, N. L. (2010). On overfitting in model selection and subsequent selection bias in performance evaluation. *Journal of Machine Learning Research*, 2079-2107.

Chang, N.-B., & Bai, K. (2017). *Multisensor Data Fusion and Machine Learning for Environmental Remote Sensing.* Boca Raton: CRC Press, Tailor and Francis Group.

Chatterjee, R., Datta, A., & Sanyal, D. (2019). Ensemble Learning Approach to Motor Imagery EEG Signal Classification. In *Machine Learning in Bio-Signal Analysis and Diagnostic Imaging* (pp. 183-208). Academic Press.

Chen, Y.-Y., Lin, Y.-H., Kung, C.-C., Chung, M.-H., & Yen, I.-H. (2019). Design and Implementation of Cloud Analytics-Assisted Smart Power Meters Considering Advanced Artificial Intelligence as Edge Analytics in Demand-Side Management for Smart Homes. *Sensors, 19*(9), 20-47.

Cortes, C., & Vapnik, V. N. (1995). Support-Vector Networks. *Journal of Machine Learning Research, 20*(3), 273-297.

Cuadra, L., Salcedo-Sanza, S., Nieto-Borge, J. C., Alexandre, E., & Rodriguez, G. (2016). Computational intelligence in wave energy: Comprehensive review and case study. *Renewable and Sustainable Energy Reviews, 58*, 1223–1246.

da Silva, I. N., Spatti, D. H., Flauzino, R. A., Bartocci Liboni, L. H., & dos Reis Alves, S. F. (2017). Artificial Neural Network Architectures and Training Processes. In *Artificial Neural Networks* (pp. 21-28). Springer,Cham.

Debeer, J., Nolte, F., & Lord, C. W. (2015). Precooking tuna: A study of tire factors impacting the time required for precooking. *Food Protection Trends, 35*, 448-460.

Demir, M., McNeese, N. J., & Cooke, N. J. (2019). The Evolution of Human-Autonomy Teams in Remotely Piloted Aircraft Systems Operations. *Frontiers in Communication, 4*, 1-12.

Department, F. F. (2012). *The State of the World Fisheries and Aquaculture*. (United Nation Food and Agriculture Organization, Fisheries and Aquaculture Department) Retrieved from http://www.fao.org/docrep/016/i2727e/i2727e00.htm

Dhar, V. (2013, December). Data Science and Prediction. *Communications of he ACM, 56*, 64-73.

Donida Labati, R., Genovese, A., Munoz, E., Piuri, V., Scotti, F., & Sforza, G. (2016). Computational intelligence for industrial and environmental applications. *IEEE 8th International Conference on Intelligent Systems (IS).* Sofia.

Education, I. O. (2017, July). *R Library Contrast Coding Systems for Categorical Variables*. (UCLA) Retrieved from stats.idre.ucla.edu/r/library/r-library-contrast-coding- systems-for-categorical-variables/

Eshwein, M. A. (2019). *How Technologies Will Change the Way Finance Departments Work – A Target Picture and Guidelines for Digital Finance.* Mannheim: Universität Duisburg-Essen.

Fausett, L. (1994). *Fundamentals of Neural Networks.* New York: Prentice Hall.

Fischetti, M., & Fraccaro, M. (2019). Machine learning meets mathematical optimization to predict the optimal production of offshore wind parks. *Computers and Operations Research, 106*, 289–297.

Fish Canning. (2020, 07 29). Retrieved from Incognito Inventions: http://incognitoinventions.com/2020/07/writing-task-1-fish-canning/

Fisher, R. A. (1921). On the "Probable Error" of a Coefficient of Correlation Deduced from a Small Sample. *Metron, 1*, 3-32.

Freedman, D. A. (2009). *Statistical Models: Theory and Practice.* Cambridge University Press.

Freund, Y., & Mason, L. (1999). The alternating decision tree learning algorithm. *Proceeding of the Sixteenth International Conference on Machine Learning.* San Francisco.

Friedman, J., Hastie, T., & Tibshirani, R. (2000). Additive Logistic Regression: A Statistical View of Boosting. *The Annals of Statistics, 28*, 337-407.

Gelman, A. (2005). Analysis of variance--why it is more important than ever. *Annals of Statistics, 33*(1), 1-33.

Ghoddusi, H., & Rafizadeh, N. (2019). Machine learning in energy economics and finance: A review. *Energy Economics, 81*, 709-727.

Glen, S. (2015). (StatisticsHowTo.com: Elementary Statistics for the rest of us!) Retrieved from Q Q Plots: Simple Definition & Example: https://www.statisticshowto.com/q-q-plots/

Grover, P. (2018). *5 Regression Loss Functions All Machine Learners Should Know*. (Heartbit) Retrieved from https://heartbeat.fritz.ai/5-regression-loss-functions-all-machine-learners-should-know-4fb140e9d4b0

Guo, B., & Yuan, Y. (2017). A comparative review of methods for comparing means using partially paired data. *Statistical Methods in Medical Research, 26*(3), 1323-1340.

Harding, J. A., Shahbaz, M., & Srinivas, A. K. (2006). Data Mining in Manufacturing: A Review. *The American Society Of Mechanical Engineering, 128*(4), 969-976.

Hastie, T., Tibshirani, R., & Friedman, J. (2008). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* New York: Springer.

Hey, T. (2009). *The Fourth Paradigm: Data-intensive Scientific Discovery.* California: Microsoft Research.

Himmetoglu, B. (2017). Retrieved from Stacking Models for Improved Predictions: https://www.kdnuggets.com/2017/02/stacking-models-imropved-predictions.html

Himmetoglu, B. (2017). *Stacking Models for Improved Predictions*. Retrieved from https://www.kdnuggets.com/2017/02/stacking-models-imropved-predictions.html

Ho, T. (1998). The random subspace method for constructing decision forests. *IEEE Tran on Pattern Anal. Mach. Intell. , 20*, 832–844.

Ho, T. K. (1995). Random Decision Forests. *Proceedings of the 3rd International Conference on Document Analysis and Recognition.* Montreal, Quebec.

Holbrook, R. (2021). *What Is Feature Engineering.* Retrieved from https://www.kaggle.com/ryanholbrook/what-is-feature-engineering

Holmes, A., Illowsky, B., & Dean, S. (2020). *Introductory Business Statistics.* Houston: openstax.

Hosmer, D., & Lemeshow, S. (1989). *Applied Logistic Regression.* John Wiley and Sons, Inc.

Huang, H.-Z., Wang, H.-K., Li, Y.-F., Zhang, L., & Liu, Z. (2015). Support vector machine based estimation of remaining useful life: current research status and future trends. *Journal of Mechanical Science and Technology, 29*(1), 151-163.

Irani, K., Cheng, J., Fayyad, U., & Qian, Z. (1993). Applying machine learning to semiconductor manufacturing. *IEEE Expert, 8*(1), 41-47.

Jacob, L., Obozinski, G., & Vert, J.-P. (2009). Group Lasso with Overlap and Graph LASSO. *Proceedings of the 26th International Conference on Machine Learning.* Montreal, Canada.

Jaensch, F., Csiszar, A., Scheifele, C., & Verl, A. (2018). Digital Twins of Manufacturing Systems as a Base for Machine Learning. *International Conference on Mechatronics and Machine Vision in Practice.* Stuttgart.

Jaradat, M., Keating, C., & Bradley, J. (2014). A histogram analysis for system of systems. *International Journal of System of Systems Engineering, 5*(3).

Jaskowiak, P. A., & Campello, R. J. (2011). Comparing Correlation Coefficients as Dissimilarity Measures for Cancer Classification in Gene Expression Data. *Brazillian Symposium on Bioinformatics.*

John, G. H., & Langley, P. (1995). Estimating continuous distributions in Bayesian classifiers. *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence.*

Kanawaday, A., & Sane, A. (2017). Machine learning for predictive maintenance of industrial machines using IoT sensor data. *IEEE International Conference on Software Engineering and Service Sciences.* Beijing.

Kanga, Z., Catalb, C., & Tekinerdogan, B. (2020). Machine learning applications in production lines: A systematic literature review. *Computers and Industrial Engineering, 149*, 106773.

Kégl, B. (2013). *The return of AdaBoost. M.H.: Multi-class Hamming trees.*

Keppel, G., & Zedeck, S. (1989). *Data analysis for research designs: Analysis of variance and multiple regression/correlation approaches.* New York: W.H. Freeman and Company.

Khmaissia, F., Frigui, H., Sunkara, M., Jasinski, J., Martinez Garcia, A., Pace, T., & Menon, M. (2018). Accelerating band gap prediction for solar materials using feature selection and regression techniques. *Computational Materials Science, 147*, 304–315.

Kireeva, N., Baskin, I., Gaspar, H., Horvath, D., Marcou, G., & Varnek, A. (2012). Generative Topographic Mapping (GTM): Universal Tool for Data Visualization, Structure-Activity Modeling and Dataset Comparison. *molecular informatics, 31*(3-4), 301-312.

Koh, H., & Tan, G. (2005). Data mining applications in healthcare. *J Healthc Inf Manag, 19*(2), 64-72.

Kohavi, R. (1995). The power of decision tables. *Proceedings of the 8th European Conference on Machine Learning, ECML '95.* Springer-Verlag, London, UK.

Kohn, M. S., Sun, J., Knoop, S., Shabo, A., Carmeli, B., Sow, D., . . . Rapp, W. (2014). IBM's Health Analytics and Clinical Decision Support. *Yearbook of Medical Informatics*, 154-162.

Krishna, S., Wan, Z., Bharadwaj, P., Whatmough, P., Faust, A., Neuman, S., . . . Reddi, V. (2021). Machine Learning-Based Automated Design Space Exploration for Autonomous Aerial Robots. *arXiv, 3*(4).

Kudyba, S. (2014). *Big Data, Mining, and Analytics.* New York: Auerbach Publications.

Kumar, E. P. (2021). *Step by Step Process of Feature Engineering for Machine Learning Algorithms in Data Science.* Retrieved 2021, from https://www.analyticsvidhya.com/blog/2021/03/step-by-step-process-of-feature-engineering-for-machine-learning-algorithms-in-data-science/

Lakshmanan, V., Gilleland, E., McGovern, A., & Tingley, M. (2015). *Machine Learning and Data Mining Approaches to Climate Science.* Springer.

Landwehr, N., Hall, M., & Frank, E. (2005). Logistic model trees. *Machine Learning, 59*, 161–205.

Lantz, B. (2013). *Machine Learning with R.* Packt Publishing Limited, ISBN - 978-1782162148.

Lavelli, A., Califf, M. E., Ciravegna, F., Freitag, D., Giuliano, C., Kushmerick, N., . . . Ireson, N. (2008). Evaluation of machine learning-based information extraction algorithms: criticisms and recommendations. *Language Resources and Evaluation, 42*, 361–393.

Lee, M. (1993). The knowledge-based factory. *Artificial Intelligence in Engineering, 8*(2), 109-125.

Liakos, K., Busato, P., Moshou, D., Pearson, S., & Bochtis, D. (2018). Machine Learning in Agriculture: A Review. *Sensors, 18*(8).

Liaw, A. (2013). *Documentation for R package randomForest.*

Linacre, J. (2008). The Expected Value of a Point-Biserial (or Similar) Correlation. *Rasch Measurement Transactions, 22*(1), 1154.

Linden, G., Smith, B., & York, J. (2003). Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Compute, 7*, 76–80.

Loukides, M. (2020, April 3). Retrieved from What is data science?: https://www.oreilly.com/radar/what-is-data-science/

Luo, C. (2021). Internet enterprise organization strategy based on FPGA and machine learning. *Microprocessors and Microsystems, 81*(0141-9331), 103714.

Lutz, W., Deisenhofer, A., Rubel, J., Bennemann, B., Giesemann, J., Poster, K., & Schwartz, B. (2021). Prospective evaluation of a clinical decision support system in psychological therapy. *Journal of Consulting and Clinical Psychology, 4*(3).

Maganathan, T., Senthilkumar, S., & Balakrishnan, V. (2020). Machine Learning and Data Analytics for Environmental Science: A Review, Prospects and Challenges. *IOP Conference Series: Materials Science and Engineering.* Tamil Nadu.

Makkonen, L. (2008). Bringing closure to the plotting position controversy. *Communications in Statistics - Theory and Methods, 37*, 460-467.

Marranzino, A. (2018). *How Machine Learning Reveals the True Costs of High-Seas Fishing*. (Massive Science) Retrieved from https://massivesci.com/articles/international-fishing-management-practices-tracked/

Marsaglia, G., Tsang, W. W., & Wang, J. (2003). Evaluating Kolmogorov's Distribution. *Journal of Statistical Software, 8*(18), 1-4.

Min, Q., Lu, Y., Liu, Z., Su, C., & Wang, B. (2019). Machine Learning based Digital Twin Framework for Production Optimization in Petrochemical Industry. *International Journal of Information Management, 49*, 502-519.

Mohd Ali, M., Basahr, A., Rabbani, M., & Abdulla, Y. (2020). Transforming Business Decision Making with Internet of Things (IoT) and Machine Learning (ML). *Decision Aid Sciences and Application (DASA).* Kingdom of Bahrain.

Myers, J. L., & Well, A. D. (2003). *Research design and statistical analysis (2nd ed).* Mahwah, NJ: Lawrence Erlbaum Associates.

Ni, D., Xiao, Z., & Lim, M. K. (2020). A systematic review of the research trends of machine learning in supply chain management. *International Journal of Machine Learning and Cybernetics, 11*, 1463–1482.

O'Driscoll, A., Daugelaite, J., & Sleator, R. (2013). big data, hadoop and cloud computing in genomics. *Biomed Inform, 46*(5), 774-781.

Onsree, T., & Tippayawong, N. (2021). Machine learning application to predict yields of solid products from biomass torrefaction. *Renewable Energy, 167*, 425-432.

Oztuna, D., Elhan, A. H., & Tuccar, E. (2006). Investigation of four different normality tests in terms of type-I error rate and power under different distributions. *Turkish Journal of Medical Science*, 171-176.

Padarian, J., Minasny, B., & McBratney, A. B. (2020). Machine learning and soil sciences: a review aided by machine learning tools. *SOIL, 6*(1), 35-52.

Park, M., & Kim, S. (2017). A review of deep learning in image recognition. *International Conference on Computer Applications and Information Processing Technology.* Bali.

Peng, J., Xiao, C., & Li, Y. (2021, 06 22). *RP2K: A Large-Scale Retail Product Dataset for Fine-Grained Image Classification.* Retrieved from https://arxiv.org/abs/2006.12634

Pérez-Martín, R. I., Banga, J. R., Sotelo, M. G., Aubourg, S. P., & Gallardo, J. M. (1989). Prediction of precooking times for albacore (Thunnus alalunga) by computer simulation. *Journal of Food Engineering, 10*(2), 83-95.

Pfanzagl, J., & Sheynin, O. (1996). Studies in the history of probability and statistics XLIV A forerunner of the t-distribution. *Biometrika*, 891–898.

Piatetsky-Shapiro, G. (1999). The Data Mining Industry Coming of Age. *IEEE Intelligent Systems and their Applications, 14*(6), 32-34.

Piryonesi, S. M., & El-Diraby, T. E. (2020). Role of Data Analytics in Infrastructure Asset Management: Overcoming Data Size and Quality Problems. *Journal of Transportation Engineering, Part B: Pavements, 146*(2), 04020022.

Praveen, V., Delhi Narendran, T., Pavithran, R., & Thirumalai, C. (2017). Data analysis using box plot and control chart for air quality. *" 2017 International Conference on Trends in Electronics and Informatics (ICEI).*

Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. (2007). *Numerical Recipes: The Art of Scientific Computing.* New York: Cambridge University Press.

Pruneau, C. (2017). The Multiple Facets of Correlation Functions. *Data Analysis Techniques for Physical Scientists*, 526-576.

Puth, M.-T., Neuhäuser, M., & Ruxton, G. D. (2014). Effective use of Pearson's product–moment correlation coefficient. *Animal Behaviour, 93*, 183-189.

Quinlan, J. (1993). *C4. 5: Programs for Machine Learning.* Morgan Kaufmann.

Quinlan, J. R. (1992). Learning with Continuous Classes. *World Scientific.*

Ranganathan, P., & Gogtay, N. (2019, July). An Introduction to Statistics - Data Types, Distributions and Summarizing Data. *Indian J Crit Care Med.*, 169-174. Retrieved from www.mymarketresearchmethods.com/types-of-data-nominal-ordinal-interval-ratio/

Rangel-Merino, A., López-Bonilla, J. L., & Linares y Miranda, R. (2005). Optimization Method based on Genetic Algorithms. *Apeiron, 12*, 393-408.

Ranstam, J., & Cook, J. (2018). LASSO regression. *British Journal of Surgery, 105*(10), 1348.

Razaviarab, N., Sharifi, S., & Banadaki, Y. M. (2019). Smart additive manufacturing empowered by a closed-loop machine learning algorithm. *Proc. SPIE 10969, Nano-, Bio-, Info-Tech Sensors and 3D Systems III.* Denver.

Reddy, C., & Aggarwal, C. (2015). *Healthcare Data Analytics.* by Chapman and Hall/CRC.

Ren, L., Cui, J., Sun, Y., & Cheng, X. (2017). Multi-bearing remaining useful life collaborative prediction: A deep learning approach. *Journal of Manufacturing Systems, 43*(2), 248-256.

Ren, R., Hung, T., & Tan, K. C. (2018). A Generic Deep-Learning-Based Approach for Automated Surface Inspection. *IEEE Transactions on Cybernetics, 48*(3), 929-940.

Rencher, A. C., & Christensen, W. F. (2012). *Methods of Multivariate Analysis.* John Wiley and Sons.

Riad, A. M., Elminir, H. K., & Elattar, H. M. (2010). Evaluation of Neural Networks in the Subject of Prognostics As Compared To Linear Regression Model. *International Journal of Engineering & Technology IJET-IJENS, 10*(6).

Ritchie, H., & Roser, M. (2019). *Seafood Production*. (Our World in Data) Retrieved from https://ourworldindata.org/seafood-production?utm_source=newsletter&utm_medium=email&utm_campaign=food&utm_content=2020-12-03

Ruder, S. (2017). An overview of gradient descent optimization algorithms . arXiv.org.

Saleem, H., Muhammad, K., Nizamani, A., Saleem, S., & Butt, J. (2021). Data Science and Machine Learning Approach to Improve e-Commerce Sales Performance on Social Web. *International Journal of Advanced Research in Engineering and Technology, 12*(4), 401-424.

Sarangi, S., Sahidullah, M., & Saha, G. (2020). Optimization of data-driven filterbank for automatic speaker verification. *Digital Signal Processing*, 102795.

Schlossberg, J. (2018). *One Fish, Two Fish: How Machine Learning Can Enhance Fisheries Management*. (Harvard Business School) Retrieved from https://digital.hbs.edu/platform-rctom/submission/one-fish-two-fish-how-machine-learning-can-enhance-fisheries-management

Schmidhuber, J. (2015). Deep Learning in Neural Networks: An Overview. *Neural Networks, 61*, 85-117.

Shayea, A., & Kadhim, Q. (2011). Artificial Neural Networks in Medical Diagnosis. *IJCSI International Journal of Computer Science Issues, 8*(2).

Sivanandam, S., & Deepa, S. (2008). *Introduction to Genetic Algorithms* . Berlin: Springer, Berlin, Heidelberg .

Sorensen, L. C., Andersen, R. S., Schou, C., & Kraft, D. (2018). Automatic parameter learning for easy instruction of industrial collaborative robots. *IEEE International Conference on Industrial Technology (ICIT).* Lyon.

Stangor, C. (2011). *Research methods for the behavioral sciences.* Belmont: Wadsworth Cengage Learning.

Stigler, S. M. (1986). *The History of Statistics: The Measurement of Uncertainty before 1900.* Cambridge, MA: Belknap Press of Harvard University Press.

Stigler, S. M. (1989). Francis Galton's account of the invention of correlation. *Statistical Science, 4*(2), 73-79.

Subramanian, A. (2020). (Seebo) Retrieved from Machine Learning and AI in Manufacturing: https://www.seebo.com/machine-learning-ai-manufacturing/

Sumner, M. F., & Hall, M. (2005). Speeding up logistic model tree induction. *Springer.*

Tamm, E. E. (2020, 06 10). Data Science in Seafood Production.

This Fish. (2019, 11 24). Retrieved from This Fish Inc | Seafood Traceability Software: https://this.fish

Thomas, M. (2020). Retrieved from 15 Examples of Machine Learning in Healthcare That Are Revolutionizing Medicine: https://builtin.com/artificial-intelligence/machine-learning-healthcare

Tibshirani, R. (1996). Regression Shrinkage and Selection via the Lasso. *Royal Statistical Society, 58*, 267-288.

Vafeiadis, T., Diamantaras, K., Sarigiannidis, G., & Chatzisavvas, K. (2015). A comparison of machine learning techniques for customer churn prediction. *Simulation Modelling Practice and Theory, 55*, 1-9.

Van Der Laan, M. P. (2007). Super learner. *Statistical Applications in Genetics and Molecular Biology, 6*(1).

Vapnik, V. N. (1995). *The Nature of Statistical Learning Theory.* Springer.

von Eye, A., & Clogg, C. C. (1996). *Categorical variables in developmental research: Methods of analysis.* Elsevier.

Vrbik, J. (2018). Small-Sample Corrections to Kolmogorov-Smirnov Test Statistic. *Pioneer Journal of Theoretical and Applied Statistics, 15*(1-2), 15-23.

Wan, Z., Xia, X., Lo, D., & Murphy, G. C. (2019). How does Machine Learning Change Software Development Practices? *IEEE Transactions on Software Engineering, 10.1109/TSE.2019.2937083*.

Wang, Y., & Witten, I. (1997). Induction of model trees for predicting continuous classes. *Proceedings of European Conference on Machine Learning Poster Papers.* Prague, Czech Republic.

Weher, E., & Allen, L. (1976). *An introduction to linear regression and correlation.* San Francisco: H. Freeman and Comp., San Francisco 1976.

Weichert, D., Link, P., Stoll, A., Rüping, S., Ihlenfeldt, S., & Wrobel, S. (2019). A review of machine learning for the optimization of production processes. *The International Journal of Advanced Manufacturing Technology, 104*, 1889–1902.

Weimer, D., Scholz-Reiter, B., & Shpitalni, M. (2016). Design of deep convolutional neural network architectures for automated feature extraction in industrial inspection. *CIRP Annals - Manufacturing Technology, 65*, 417–420.

Wendl, M. C. (2016). Pseudonymous fame. *Science, 351*(1406).

Wenzel, H., Smit, D., & Sardesai, S. (2019). A literature review on machine learning in supply chain management. *Proceedings of the Hamburg International Conference of Logistics.* Hamburg.

Wigley, P. B., Everitt, P. J., van den Hengel, A., Bastian, J. W., Sooriyabandara, M. A., McDonald, G. D., . . . Hush, M. R. (2016). Fast machine-learning online optimization of ultra-cold-atom experiments. *Scientific Reports, 6*, 25890.

Wilkinson, L., & Friendly, M. (2012). The History of the Cluster Heat Map. *The American Statistician , 63*(2), 179-184.

Witten, I., & Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques.* Morgan Kaufmann Publication.

Wolpert, D. (1992). Stacked Generalization. *Neural Networks, 5*, 241-259.

Wrobel, D., Weichert, P., Stoll, A., Ruping, S., Ihlenfeldt, S., & Wrobel, S. (2019). A review of machine learning for the optimization of production processes. *The International Journal of Advanced Manufacturing Technology, 104*, 1889–1902.

Xie, Y., Honbo, D., Choudhary, A., Zhang, K., Cheng, Y., & Agrawal, A. (2012). Voxsup: A social engagement framework. *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD).*

Yan, X., & Su, X. G. (2009). *Linear Regression Analysis: Theory and Computing.* World Scientific.

Yue, W., Wang, Z., Chen, H., Payne, A., & Liu, X. (2018). Machine Learning with Applications in Breast Cancer Diagnosis and Prognosis. *MDPI, 2*(2).

Zhang, T. (2004). Statistical behaviour and consistency of classification methods based on convex risk minimization. *Annals of Statistics, 32*(1), 56-85.

Appendices

Appendix A. Database Metadata

Process Control Parameters and Raw Material Parameters are shown in Table 53.

Table 53 List of Process Control and Raw Material Parameters

Variable Name	Data Field Name	Description	Unit
Species	SPEC / lot.code.SPEC	AL = Albacore; YF = Yellow Fin. BE = Big Eye; SK = Skipjack. BT = Bonito; TG = Tonggol.	Blank-No Unit
Raw Material Code	RM_ID_CODE / lot.code.name	Traceability code used to track raw material through factory.	Blank-No Unit
Marine Stewardship Council (MSC) Code	MSC / lot.code.MSC	Code identifying the certified sustainable fishery	Blank-No Unit
MSC Batch	MSC_Batch	Batch code for MSC product inside factory	Blank-No Unit
Trip Date Start	TD_Start	The day the fishing vessel began its fishing trip	Datetime
Trip Date End	TD_End	The day the fishing vessel ended its fishing trip	Datetime
Unloading Date	Unload_Date	Date that fish was brought onshore and unloaded at a port	Datetime
Offloading Date	Offload	Date that the fish was offloaded from the fishing vessel onto a transhipment vessel	Datetime
FAO Major Fishing Area	FAO	FAO statistical area for catch location	Blank-No Unit
Fishing Area	CT_Area	Common name for FAO statistical Area	Blank-No Unit
Fishing Method	CT_Method	Method for how the fish was caught	Blank-No Unit
Fishing Vessel Name	FV	Name of fishing vessel	Blank-No Unit
Fishing Vessel Flag State	FV_Flag	Country where the fishing vessel is legally registered	Blank-No Unit
Fishing Vessel IMO Number	FV_IMO	Registration number of vessels from the International Maritime Organization (IMO)	Blank-No Unit
Fishing Vessel Captain	Captain	Name of captain of fishing vessel	Blank-No Unit
Transhipment Vessel	MV	Carrier or transhipment vessels often collect catch from fishing vessels at sea to enable the fishing vessel to stay at sea fishing longer. The transhipment vessel will collect the catch and often replenish the fishing vessel with fuel, food and even crew.	Blank-No Unit
Transhipment Vessel Flag State	MV_Flag	Country where the transhipment vessel is legally registered	Blank-No Unit

Variable Name	Data Field Name	Description	Unit
Transhipment Vessel IMO Number	TV_IMO	Registration number of vessels from the International Maritime Organization (IMO)	Blank-No Unit
Raw Material Quantity Received	QTY_RV	Quantity received at the cold storage of the factory. KG is unit of measurement.	Blank-No Unit
Raw Material Quantity Invoiced	QTY_IV	Quantity of raw material on invoice	Blank-No Unit
Raw Material Quantity Captain Declared	QTY_CT	Quantity of catch declared by the captain	Blank-No Unit
Landing Country	Landing	Country where the fish was brought to port	Blank-No Unit
Landing Port	Port	Port where the fish was brought onshore	Blank-No Unit
Fish Broker	Broker	Company that brokered the sale of fish	Blank-No Unit
Receiver	Receiver	Company that received the fish when it was first brought onshore	Blank-No Unit
Freezing Location	Freezing	Location where the fish was frozen, i.e. on a vessel or in a factory	Blank-No Unit
Pack Date	lot.pack_date	Date of production in factory	Datetime
Out of Cold Storage Date	dump.from_cold_room_time	Time that the fish leaves the cold storage	Datetime
Size Grade	lot.size	Size of the fish	Blank-No Unit
Thaw On	thaw.thaw_on_time	Time to start thawing fish in bins of water	Datetime
Thaw Off	thaw.thaw_off_time	Time to stop thawing fish in bins of water	Datetime
Backbone (BB) Temps (after thawing)	thaw.bb_temps_c	12 BB Temp. samples after the thawing process	˚C
Thawing Water Temp.	thaw.water_temp_c	Temperature of water during thawing	˚C
Thawing Drain Time	thaw.drain_time	Time that water is drained from the bin	Datetime
Gutting Start Time	thaw.gutting_start_time	Fish are gutted - start	Datetime
Gutting End Time	thaw.gutting_finish_time	Fish are gutted - end	Datetime
Precooker Internal Temperatures	rack.internal_temps_c	6 temperature samples	˚C
Avg Precooker Internal Temperatures	rack.internal_temps_c.avg	Temperature of fish is taken prior to going into the precooker	˚C
Precooking Corrective Action	rack.corrective_action	Corrective Action; yes or no; this is a text field	Blank-No Unit
Weight of fish	rack.weights_kg	6 fish are sampled for their weight	Kg

Variable Name	Data Field Name	Description	Unit
Average weight of one fish	rack.weights_kg.avg	Dependent Variable: 6 fish are sampled for their weights	kg
Precooking Steam On Time	precook.steam_on_time	Turning on the steam on the precooker	Datetime
Precooking Start Time	precook.cook_start_time	Starting the cooking process	Datetime
Precooking Steam Off Time	precook.steam_off_time	Turning off the steam on the precooker; fish continue to cook	Datetime
Precooker Temperature	precook.cooker_temp_c	Temperature on the	˚C
Precooking Time	precook.cooking_time_min	Total time for cooking the fish	minutes
Precooking Backbone (BB) Temperatures	precook.bb_temps_c	24 samples taken	˚C
Temp < 60°C Extra Time	precook.extra_minutes	If the fish isn't properly cooked, they will cook it more and add more minutes of cooking	minutes
Extra Steam Off Time	precook.extra_steam_off_time	Extra time if the fish isn't properly cooked	Datetime
Extra BB Temps	precook.extra_bb_temps_c	24 samples taken	˚C
Spray Time Start	precook.spray_start_time	Spraying the fish with cool water to moisten and cool the fish prior to processing	Datetime
Spray Time End	precook.spray_stop_time	Spraying the fish with cool water to moisten and cool the fish prior to processing	Datetime
After Spraying Temperature	precook.after_spray_temps_c	6 samples	˚C
Chilling Start	chill.start	Cooked fish put in chilling room with mist	Datetime
Chilling Room Temperature	chill.room_temp_c	Cooked fish put in chilling room with mist	˚C
Cleaning Start Time	clean.cleaning_start_time	Time that workers start to clean the meat off the cooked tuna	Datetime
Cleaning BB Temperature	clean.bb_temps_time	6 temperature samples collected	˚C
BB After Chilling Temps	clean.bb_after_chill_temps_c	Dependent Variable: calculated from 6 samples	˚C
Cleaning Flake to Loin Ratio (Set-up)	clean.setup.ratio	Targeted ratio of flakes to loin	Blank-No Unit
Cleaning Corrective Action (Stop, Inform)	clean.corrective_action	Corrective action; yes or no; this is a text field	Blank-No Unit
Targetted fill weight (g)	pack.setup.fill_weight_g	Targeted fill weight of meat in the can	g
Can Weight Sample	can.weights_g	20 samples weights taken of the can	g

Variable Name	Data Field Name	Description	Unit
Can Weight Average	can.weights_g.avg	Dependent Variable: Average weight calculated from 20 samples	g
Can Target Weight	can.target_weight	Targeted weight of can	g
Targeted Net Weight of Meat	pack.setup.net_weight_g	This is the targeted amount of fish to put in the can	g
Can Code	pack.setup.can_code	Once the fish is put into a can, the cans are traceable by Can Code which is linked to the Raw Material Code	Blank-No Unit
Can Net Weight / Fill Weight	can.nw_fw	Ratio of net weight to fill weight	g
Can Size	data.can_size	Size of the can used for the product; related to product type	Blank-No Unit
Seaming First Can Closed Time	seam.first_can_close_time	Time that the first can is seamed	Datetime
Packing Start Time	pack.start_time	Time that the lot begins packing into can	Datetime
Packing Finish Time	pack.finish_time	Time that the lot ends packing into can	Datetime
Total RM Ton	rm_ton_total	Total amount of raw material from lot used in production	Kg
Fill Weight	fill_weight	Amount of mean put into the can	g
Can Count	can_count	Number of cans created	Blank-No Unit
Yield FW (%)	yield_fw	Target Variable: Yield is calculated by taking the Fill Weight multiplied by the number of cans to get the total Meat used and then divided by the Raw Material Tons	Blank-No Unit

Appendix B. Objects in Process Control Parameter Dataset

The structure of objects in the Process Control Parameter Dataset is presented in Table 54.

Table 54 Process Parameters Objects

	Columns Names	Names of Objects	Name of Sub-Objects	Type of Every Object
1	pk	*N/A*	*N/A*	
2	user	*N/A*	*N/A*	
3	dump	Dump	*N/A*	string of 1 element (1 element of string type)
		weight_kg	*N/A*	number
		from_cold_room_time	*N/A*	string of 1 element
4	rack	total_pans	*N/A*	number
		weights_kg	avg	number
			max	number
			min	number
			values	string of 6 elements (vector of 8 string elements)
		fish_per_pan	*N/A*	number
		pans_per_rack	avg	number
			max	number
			min	number
			values	string of 10 elements
		internal_temps_c	avg	number
			max	number
			min	number
			values	string of 6 elements
5	retort	Batch	*N/A*	string of 1 element
		Setup	pk	string of 1 element
			mig	string of 1 element
			retort	string of 1 element
		Retort	*N/A*	string of 1 element
		mig_temp_c	*N/A*	number
		rt_operator	*N/A*	string of 1 element
		chart_temp_c	*N/A*	number
		chlorine_ppm	*N/A*	number
		gauage_kg_cm2	*N/A*	number
		temp_up_time	*N/A*	string of 1 element
		steam_on_time	*N/A*	string of 1 element
		steam_off_time	*N/A*	string of 1 element
		initial_temps_c	avg	number
			max	number
			min	number
			values	string of 5 elements
		vent_close_time	*N/A*	string of 1 element
		drain_close_time	*N/A*	string of 1 element
		basket_load_style	*N/A*	string of 1 element
		tag_colour_change	*N/A*	string of 1 element
		vent_close_temp_c	*N/A*	number
		basket_unload_time	*N/A*	string of 1 element
		drain_close_temp_c	*N/A*	number
		proc_cond_gauge_bar	values	string of 8 elements
		basket_unload_temp_c	*N/A*	number
		proc_cond_flow_l_sec	values	string of 8 elements

	Columns Names	Names of Objects	Name of Sub-Objects	Type of Every Object
		proc_cond_mig_temp_c	values	string of 8 elements
		top_bleeder_observed	*N/A*	string of 1 elements
		chlorine_checked_time	*N/A*	string of 1 element
		cond_bleeder_observed	*N/A*	string of 1 element
		proc_cond_chart_temp_c	values	string of 8 elements
		schedule_process_temp_c	*N/A*	number
		schedule_process_time_min	*N/A*	number
		main_steam_pressure_kg_cm2	*N/A*	number
6	pack	Line	*N/A*	string of 1 element
		Setup	pk	string of 1 element
			name	string of 1 element
			brand	string of 1 element
			o_l_min	number
			species	string of 1 element
			t_r_min	number
			water_g	number
			can_code	string of 1 element
			can_size	string of 1 element
			lid_type	string of 1 element
			maxfill_g	number
			pack_date	string of 1 element
			can_line_1	string of 1 element
			can_line_2	string of 1 element
			water_type	string of 1 element
			butting_min	number
			can_supplier	string of 1 element
			lid_supplier	string of 1 element
			net_weight_g	number
			packing_line	string of 1 element
			fill_weight_g	number
			packing_line_run	string of 1 element
		start_time	*N/A*	string of 1 element
		finish_time	*N/A*	string of 1 element
7	seam	Line	*N/A*	string of 1 element
		basket	line	string of 1 element
			number	number
			can_count	number
		num_baskets	*N/A*	number
		multi_group_name	*N/A*	string of 1 element
		basket_start_time	*N/A*	string of 1 element
		basket_finish_time	*N/A*	string of 1 element
		first_can_close_time	*N/A*	string of 1 element
8	lot	Pk	*N/A*	string of 1 element
		Code	CC	string of 1 element
			FV	string of 1 element
			MV	string of 1 element
			FAO	string of 1 element
			IMD	string of 1 element
			MSC	string of 1 element
			Ref	string of 1 element

	Columns Names	Names of Objects	Name of Sub-Objects	Type of Every Object
			MCPD	string of 1 element
			Port	string of 1 element
			RMBS	string of 1 element
			SPEC	string of 1 element
			name	string of 1 element
			Note1	string of 1 element
			Port2	string of 1 element
			TOTAL	number
			Wharf	string of 1 element
			Broker	string of 1 element
			EXP_9M	string of 1 element
			FV_IMO	string of 1 element
			QTY_CC	number
			QTY_CT	number
			QTY_IV	number
			QTY_RV	number
			Status	number
			TD_End	string of 1 element
			TV_IMO	string of 1 element
			CT_Area	string of 1 element
			Captain	string of 1 element
			Code_RM	string of 1 element
			EXP_12M	string of 1 element
			FV_Flag	string of 1 element
			Invoice	string of 1 element
			Landing	string of 1 element
			MV_Flag	string of 1 element
			Offload	string of 1 element
			QTY_OUT	number
			Freezing	string of 1 element
			Receiver	string of 1 element
			Roworder	number
			TD_Start	string of 1 element
			CT_Method	string of 1 element
			Container	string of 1 element
			MSC_Batch	string of 1 element
			Transport	string of 1 element
			RM_ID_CODE	string of 1 element
			Unload_Date	string of 1 element
		Name	*N/A*	string of 1 element
		Size	*N/A*	string of 1 element
		pack_date	*N/A*	string of 1 element
		cs_facility	*N/A*	string of 1 element
9	clean	Line	*N/A*	string of 1 element
		Setup	pk	string of 1 element
			name	string of 1 element
			ratio	string of 1 element
			cleaning_line	string of 1 element
		md_test	*N/A*	string of 1 element
		rack_order	*N/A*	string of 1 element
		trolly_code	*N/A*	string of 1 element
		bb_temps_time	*N/A*	string of 1 element
		pans_per_rack	*N/A*	string of 1 element
		md_test_eval_calc	*N/A*	string of 1 element
		cleaning_start_time	*N/A*	string of 1 element
		bb_after_chill_temps_c	avg	number
			max	number

	Columns Names	Names of Objects	Name of Sub-Objects	Type of Every Object
			min	number
			values	string of 6 elements
10	precook	Batch	*N/A*	string of 1 element
		cooker	*N/A*	string of 1 element
		bb_temps_c	avg	number
			max	number
			min	number
			values	string of 24 elements
		cooker_temp_c	*N/A*	number
		steam_on_time	*N/A*	string of 1 element
		racks_in_batch	*N/A*	string of 1 element
		steam_off_time	*N/A*	string of 1 element
		cook_start_time	*N/A*	string of 1 element
		spray_stop_time	*N/A*	string of 1 element
		cooking_time_min	*N/A*	number
		extra_bb_temps_c	avg	number
			max	number
			min	number
			values	string of 24 elements
		spray_start_time	*N/A*	string of 1 element
		after_spray_temps_c	avg	number
			max	number
			min	number
			values	string of 6 elements