Predicting Coral Bleaching events using climatic factors along with ENSO cycles

Tanya Sabarwal
201799811

Supervised by Dr. Sam Pegler

Submitted in accordance with the requirements for the
module MATH5872M: Dissertation in Data Science and Analytics
as part of the degree of

Master of Science in Data Science and Analytics

The University of Leeds, School of Mathematics

September 2024

UNIVERSITY OF LEEDS

School of Mathematics

FACULTY OF ENGINEERING AND PHYSICAL SCIENCES

Academic integrity statement

I am aware that the University defines plagiarism as presenting someone else's work, in whole or in part, as your own. Work means any intellectual output, and typically includes text, data, images, sound or performance.

I promise that in the attached submission I have not presented anyone else's work, in whole or in part, as my own and I have not colluded with others in the preparation of this work. Where I have taken advantage of the work of others, I have given full acknowledgement. I have not resubmitted my own work or part thereof without specific written permission to do so from the University staff concerned when any of this work has been or is being submitted for marks or credits even if in a different module or for a different qualification or completed prior to entry to the University. I have read and understood the University's published rules on plagiarism and also any more detailed rules specified at School or module level. I know that if I commit plagiarism I can be expelled from the University and that it is my responsibility to be aware of the University's regulations on plagiarism and their importance.

I re-confirm my consent to the University copying and distributing any or all of my work in any form and using third parties (who may be based outside the EU/EEA) to monitor breaches of regulations, to verify whether my work contains plagiarised material, and for quality assurance purposes. I confirm that I have declared all mitigating circumstances that may be relevant to the assessment of this piece of work and that I wish to have taken into account. I am aware of the University's policy on mitigation and the School's procedures for the submission of statements and evidence of mitigation. I am aware of the penalties imposed for the late submission of coursework.

Name TANYA SABARWAL

Student ID 201799811

Abstract

Coral bleaching has been one of the most damaging ecological phenomena in recent years, and it has definitely been exacerbated by increased temperatures at the surface of the ocean. This thesis is particularly directed towards applying supervised machine learning models to identify those areas where coral bleaching has occurred with an attempt to also predict the percentage of bleaching at the site. A comparative study was carried out based on four classifiers, including Random Forest, Decision Tree, k-Nearest Neighbors (KNN), and Logistic Regression. Among these, the Random Forest Classifier proved to be the most efficient in classifying sites exhibiting bleaching. Using the mean importance of features obtained from random forest classifier, features with higher mean importance(greater than 0.02) were selected to further employ regression models to predict the actual bleaching percentage at any give site. The regression models were fine-tuned using GridSearchCV to identify the best hyperparameters that make the model more generalised (perform better even on unseen instances). The XGBoost regressor outperformed linear regression and random forest regressor in generalisation and reliability. But due to data issues such as heavy skewness in the target variable, varied correlation across environmental features, and inconsistencies in the data collection process, there is scope to improve the performance for regression tasks. An important aspect of climate change known as the ENSO cycle period was studied and the impact of the phases in the cycle over bleaching events was explored. The project outlines the data science cycle with steps involving data understanding, data preparation, modelling, and finally interpreting the obtained results. Lastly, along with the implications from results, a discussion on future research is done with respect to the findings.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 What is coral bleaching?

Most of us are fascinated by scuba diving. One of the purposes of scuba diving is to explore and admire the beauty of the life underwater. One such part of that life are the coral reefs. Coral reefs are formed by tiny individual organisms called coral polyps. They are not only aesthetically pleasing but are home to more than 25 percent of all marine life. Hence they are sometimes referred to as the 'rainforests of the sea' [27].They have countless other benefits such as they aid in reduction of coastal erosion, provide recreation and tourism opportunities which help local economies, source of food and new medicines [22].

Figure 1.1: Montipora coral which is normally dark brown but has turned flourescent blue due to bleaching [29].

A large number of corals live in a symbiotic relationship with an an algae called zooxanthellae which is present in the tissues of the corals. With the help of this algae corals fulfil atmost

90 percent of their energy requirements since it produces energy-rich compounds through the process of photosynthesis. It also gives the corals most of their colouration due to the photosynthetic pigments. Coral Bleaching occurs when this relationship is under stress, leading to the ejection of zooxanthellae and causing the corals to turn transparent, bright yellow or white. However, coral reefs can eventually recover from a bleaching event but it takes them as long as 15 years given that there are minimal to no disturbances in the environment since they need stable conditions to re-establish [3].

The increase in water temperatures has been the primary cause of coral bleaching from the past 20 years. This is because temperature increase of mere 1 or 2 degrees can prompt mass bleaching events. And the most significant reason for rise in temperature is climate change. For instance, on the Great Barrier Reef the average sea surface temperature has risen by more than 0.8 degrees since 1880. Furthermore, Scientists have indicated hat global bleaching events will occur annually by 2050. The intervals between bleaching events are getting shorter and shorter [16]. Since 2023, mass bleaching of coral reefs has taken place in more than 50 areas across the world. In fact, the world is currently experiencing its fourth global coral bleaching event which is the second in this decade. The first global bleaching event happened in 1998 which was followed by the most severe, widespread and longest bleaching event from 2014-2017 which was triggered by El Nino. During this period more than 75 percent of Earth's tropical reefs experienced bleaching-level heat stress, and at nearly 30 percent of reefs, it reached mortality level.

The cases of coral bleaching have been observed for nearly a century, but prior to 1980s the cases were finite in terms of area and duration. The known bleaching cases were mostly observed in areas such as small tide pools, shallow enclosed lagoons and when hot cloudless conditions led to increased temperatures. Since the 1980s, bleaching drastically increased and instead of being confined to smaller areas it was observed at huge areas of the ocean covering up thousands of kilometers. Among all the potential causes, only extreme temperatures was noticed in all cases. At present, the corals alive have little or no ability to adapt to warmer temperatures [1].

1.2 Previous research analysis

Now that we have a brief idea about coral reefs, their importance and the recent increase in the occurrence of bleaching events, let us look at few a pieces of research done in this field in the recent times. The studies outlined below also aim to address this issue using different techniques such as machine learning models, identifying factors (majorly focusing on climatic factors) contributing towards bleaching. These approaches aim to provide accurate predictions and a clear understanding of the global patterns of coral bleaching. Research in this field possesses a greater importance in today's climate and it could help formulate conservation strategies and policy decisions to protect these endangered ecosystems. Through the project, we gain a better

understanding of the data at hand while simultaneously compare our results to check whether the results we obtain hold truth in the context of predictive patterns of coral bleaching.

1.2.1 Coral bleaching predictor using Machine Learning [20]

This paper aimed to predict the likelihood of bleaching in a particular region in an attempt to aid policy makers and organisations in easier allocation of resources for conservation of endangered bleaching areas. The data is acquired from the Biological and Chemical Oceanography Data Management Office which had instances of bleaching events recorded across oceans and the parameters under which those events occurred. The authors eliminated parameters with missing values and focused on finding the most relevant parameters on which coral bleaching depends. Using methods such as linear correlation analysis and principal component analysis the least multicollinear parameters were found to be sea surface temperature, sea surface temperature anomaly, latitute, longitude and depth.

The lazypredict Python library was used to run thirty nine regression models automatically on the least multicollinear parameters found. Among regression models like Decision Tree Regressor, Passive Aggressive Regressor, Gradient Boosting Regressor and many others, Random Forest Regressor was found to be the best performing with the least R-Squared value of 0.25. Additionally, Grid Search was used to determine the most optimal hyperparameters for the random forest model. Futhermore, a K-means clustering model was implemented to cluster sites according to the risk of bleaching that is high-risk(sites with higher bleaching percentages) and low risk(sites with lower bleaching percentages).

1.2.2 Case study of Cu Lao Cham [8]

This paper calls for the need for practical action to manage the affects of climate change. The authors conducted this study on the corals present in the Cu Lao Cham islands which is also a part of the biosphere reserve in Vietnam. It was discovered that the coral cover diminished by 47 percent from the year 2004 to 2016. The reasons that led to this decline are believed to be increased temperatures due to climate change and warm events like El Nino. El Nino is a phenonmenon when sea temperatures in the tropical eastern regions of Pacific ocean get unusually warm. It takes place at unanticipated periods ranging from two to seven years. From the 1950 to 2021, 5 strong and 3 very strong el nino events took place. Coral reefs are at the risk of heat shock due these events combined with the effect of climate change.

This study generated projections of future temperatures using Proudman Oceanographic Laboratory Coastal Ocean Modelling System (POLCOMS) which is a three-dimensional hydrodynamic model that simulates the physical processes of energy and momentum transfer in the ocean; its outputs include temperature, salinity, and current speed. These projections were further validated using correlation analysis and regression. Additionally, they were also plotted against satellite based observations. Finally, it is concluded that in the future there would be

prolonged period of extreme temperature from which the corals are unlikely to recover.

1.2.3 Climate change and coral bleaching [4]

This paper highlights the impact of climate change on coral bleaching which has led to elevated mortality rates. The authors have used both supervised and unsupervised methods of machine learning to provide evidence of factors leading to coral bleaching and interpret the relationship among them. Classification algorithms such as Naive Bayes, Support vector machines (SVM) and decision trees were used to predict the condition of corals and classify them into one of the five categories being completely damaged coral, damaged coral, moderately luxuriant coral, luxuriant coral, and perfectly luxuriant coral. Among the above mentioned models, support vector machines yielded the highest accuracy of almost 89 percent. Additionally, K-means clustering algorithm is applied to determine relationship between causal factors and the condition of coral reefs. This study was conducted on dataset obtained from Department of Marine and coastal resources, Thailand during the years 2013-2018. It was concluded that factors such as ph, sea surface temperature and wind speed are highly correlated with coral bleaching. It also provides evidence that machine learning models are a viable and useful approach to monitoring and analyzing coral reef bleaching under climate change.

1.2.4 Exploring relationship among variables [12]

The aim of this thesis was to explore the relationship among variables such as depth, exposure, distance to shore, and temperature for percent bleaching. The study was conducted using two different datasets, one being from an open source repository and the other was the global bleaching dataset from the Biological and Chemical Oceanography Data Management Office. For data cleaning in both the datasets missing values were dropped. To identify relationships between the variables mentioned above with coral bleaching percentages line plots and box plots were used while also employing linear regression to examine p-values and R-squared values. To check if distance to shore is a good predictor variable for bleaching percentages, quadratic and logistic models, splines were also explored in addition to linear models on selected few eco-regions such as Bahamas and Florida Keys, Belize and West Caribbean, Hispaniola Puerto Rico and Lesser Antilles, Sulu Sea, Central and northern Great Barrier Reef. The regression model indicated a weak positive correlation between distance from shore and bleaching levels which means as the distance to shore increases, the bleaching percentages increase. But the relationship was inconsistent. A weak negative correlation was found between global temperatures and bleaching percentages, which was implausible. The R-squared value was very low, indicating that temperature alone does not explain coral bleaching adequately. Furthermore, a case study on Broward County done using the open source bleaching mortality dataset ranging from 1963 to 2017 indicated slight positive relationship between temperature and Mortality levels. But again a lower R-squared value makes the model unreliable. Potential causes of these results were due to the data being inconsistent as there were gaps within years, months and locations. Addition-

ally, data was collected from different sources which would have also contributed to the issues related to data format and quality along with a lot of missing data.

1.3 Research objective and question

The overall aim of this dissertation is to identify the key climatic factors that are contributing towards the bleaching of corals. Drawing inspiration from above mentioned research, the aim is to build a classification model to identify the important features and predict whether a coral has bleached. The results from classification will then be utilized to decide on features to train an effective regression model to accurately predict the exact bleaching percentages for coral sites that were initially classified to be bleached.

The project aims to tackle issues related to data quality and apply machine learning models for classification including logistic regression, K-Nearest Neighbours, Decision Trees and Random Forests to effectively distinguish between sites with bleaching and no bleaching. The project also aims to build on the results from classification and build regression algorithms such as Random Forest Regressor and XGBoost Regressor to determine the exact percentage of coral bleaching at any given site. The variables related to climate such as sea surface temperature, thermal stress anomaly along with other variables such as depth will be studied in order to answer the question: are factors pertaining to climate change a major factor in influencing the increase of bleaching events?

Using the knowledge of already used machine learning models from [20], [12], [4] the novelty of this project is feature engineering a new variable called 'Term-Type'. This added column provides information regarding the ENSO cycle occurring during the time of recording observations at the site. The important phases of the ENSO cycle (El-Nino and La-Nina) were discussed and with certain data operations, information from a secondary data source (ENSO cycles) was added to the primary data of recorded coral observations. This extends the aim of this project to try and explore the relationship that ENSO periods have on coral bleaching events. The addition of the column will further increase the scope of future research where coral bleaching could be understood better within the context of climate change.

Chapter 2

Dataset

2.1 Data understanding

The data used for this analysis is acquired from a reliable source the Biological and Chemical Oceanography Data Management Office (BCO-DMO). BCO-DMO curates a database of research-ready data spanning the full range of marine ecosystem-related measurements including in-situ and remotely-sensed observations, experimental and model results, and synthesis products[32]. There are two versions of this dataset available. In both the versions some processing was done. For this analysis, the most recent processed version was used which was released in October of 2022.

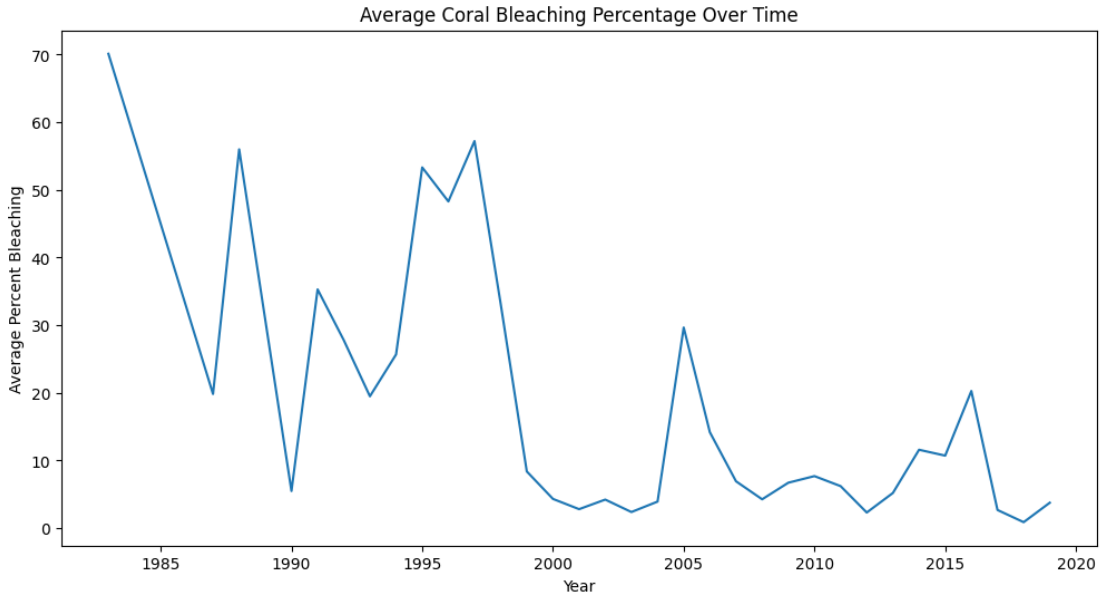


Figure 2.1: Average percentage of bleaching over the years.

Global Coral-Bleaching Database (GCBD) is available as a CSV file containing instances

obtained from seven data sources: 1) Reef Check, 2) Donner, 3) McClanahan, 4) AGRRA, 5) FRRP, 6) Safaie, and 7) Kumagai, which are spread across 5 oceans namely Pacific, Atlantic, Red sea, Arabian gulf, and Indian over 40 years, from 1980-2020. The data set consists of 62 columns and 41,361 rows. The database contains information on the presence and absence of coral bleaching—allowing comparative analyses and the determination of geographical bleaching thresholds—together with site exposure, distance to land, mean turbidity, cyclone frequency, and a suite of sea-surface temperature metrics at the times of survey.The issues with the dataset that are known to BCO-DMO is that there is little data on coral bleaching before 1998 bleaching event and most of the data were collected between 2015 and 2016. Possible explanations for these issues could be that bleaching events were not observed in such larger areas before 1998 [24]. And there is a lot of data between the years 2015-2016 due to the occurrence of a global bleaching event which lasted longer than any other bleaching event as shown in figure 2.1 .

Most of the parameters listed in the dataset are very relevant for this analysis. A few important columns are described below.

- **Distance to Shore**: This variable indicates the distance(m) of the sampling site from nearest land. This distance influences the coral reefs in different ways. Coral reefs which are at a closer proximity to land are often at a high risk of being exposed to eutrophication, land erosion, pollution, overfishing, coastal development, tourism and temperature fluctuations. On the other hand coral reefs that are further from land often provide shelter to higher biodiversity and are more resilient to environmental changes [25].
- **Turbidity**: kd490 with a 100-km buffer Turbidity is a measure of how cloudy the water is. Higher turbidity could be caused by both natural and human-induced factors such as erosion, algae, mining, construction activities, storm, factory waste, organic waste. Since increased turbidity can reduce the penetration of light it can significantly impact the process of photosynthesis and make the corals more susceptible to bleaching. However, it can be advantageous temporarily as it can protect from immediate affects of thermal stress [19].
- **Cyclone Frequency**: Climate change is leading to increased temperatures on both land and oceans which is increasing the occurrence and intensity of cyclones around the world. After such a powerful calamity, most corals are torn from their footing and debris is left behind. Though coral reefs have the capacity to naturally recover from this profound impact, it can take a more than a decade. But with the recurrence of these cyclones there is not enough time for healing [11].
- **Exposure**: This variable indicates how endangered a site is to predominate winds, swell, and fetch(extent of open ocean). A site was considered exposed if it experienced strong seasonal winds or prevailing winds had fetch greater than 20km. Otherwise, the site was considered 'Sheltered' or 'Sometimes'. 'Sometimes' indicates areas with a fetch more than 20 km through a small geographic window that may be exposed during cyclone

season [32].

- **Climatological Sea Surface Temperature (SST)**: It is the temperature of the top millimeter of the ocean's surface. Since the sea surface is the boundary between the ocean and atmosphere it can be used to interpret the flow of energy between the two. Due to this sea surface temperature has its influence in atomospheric circulation, rainfall patterns and tropical cyclones which are some of the important components of the climate system. SST for coral reefs to grow is between 22°C and 28°C. However, the temperature range in which corals can survive is between 18°C and 36°C[5].
- **Sea Surface Temperature Anomaly (SSTA)**: An anomaly in the sea surface temperature suggests a departure from average temperature conditions. A positive anomaly signals the observed temperature was warmer than the baseline, while a negative anomaly signals the observed temperature was cooler than the baseline[23].
- **SSTA_DHW**: Degree Heating Week (DHW) quantifies the combined stress experienced by coral reefs over a period of 12 weeks[31].
- **TSA**: Thermal stress refers to the stress caused by fluctuations in temperature. Thermal stress anomaly is when there is an abnormal deviation from the expected temperature conditions, which further alleviates the stress on the corals. This stress can trigger coral bleaching.
- **Depth**: This variable indicates the depth of the sampling site. Bleaching percentages are often higher in corals at shallower depths because of high exposure to sunlight and fluctuations in sea surface temperatures. In contrast, deeper corals are often situated in more thermally stable environments, which can reduce their exposure to temperature extremes and consequently lower their risk of bleaching [9].
- **Windspeed**: Strong winds can help cool sea surface temperatures (SSTs) by enhancing the mixing of surface waters with cooler, deeper waters. This process, known as upwelling, can bring cold water from below to the surface, thereby lowering SSTs in the affected area. Cooler surface waters can reduce the thermal stress on corals, making them less susceptible to bleaching [26].

2.2 Data cleaning

Even though the data is acquired from a reliable source, there were gaps in the dataset that had to be addressed to produce consistent and structured data which will aid in making informed decisions in the modelling stage. With this dataset the issues encountered were inconsistencies such as missing values, irrelevant observations and columns.

Generally, Missing values or null values can occur due to human error when processing data, machine error due to equipment malfunctioning, merging unrelated data, etc. This missing

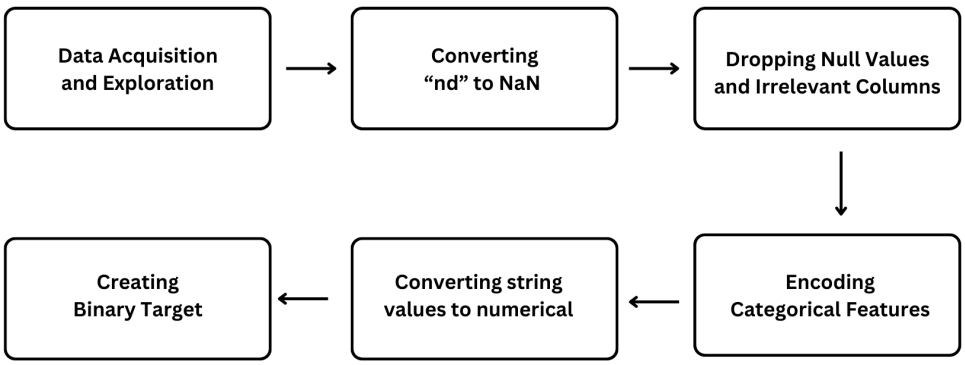


Figure 2.2: Steps followed for data preparation.

data can lead to a number of problems such as performance degradation of models, biased outcomes, inaccurate relationship among variables, skewed statistical summaries and misleading conclusions from the visualisations and analysis. Hence, they need to be handled. The most common methods of handling missing values is either dropping them from the data or imputing them based on statistical measures like mean, median and mode[30].

In this dataset, There were 4 columns in which 90 to 100 percent of the observations were missing. These columns were dropped since they did not add any value to the dataset. Additionally, missing values were also spread across the dataset in the form of a string 'nd'. To solve this issue, the string 'nd' values were first converted to NaN type which pandas uses to represent null or missing values. After this identification, Five columns were completely dropped from the data set because they had more than 10,000 missing values. The other columns in which missing values were less than 10,000, rows where the values were missing were dropped. Additionally, a few other columns such as city name, country name, province name were also dropped since they were irrelevant to the analysis. After the aforementioned changes, the dataset had 32678 rows and 47 columns.

2.3 Data processing

Now that we have a clean data with no missing values, we can further look into other aspects of it like which parameters are the most useful, the data types of columns, format of the data, etc. Preprocessing is converting raw data into useful information. It prepares the data for modelling by improving data consistency and makes it more readable for machine learning algorithms.

The dataset contains a total of 62 columns. Hence, it is necessary to identify which of these columns are the most relevant based on domain knowledge as well as correlation with the target variable. To proceed with it, it is important to further process the data.

Representation

The dataset consists of both numerical and categorical columns. Upon further examination of the columns, it was found that almost all columns containing numerical values were being represented as strings (object type) instead of int or float data type. Data type tells the kind of values the column contains. Not only does object type data take up more memory, it makes it difficult for us to perform any mathematical operations or aggregating functions on the data. Furthermore, it is an incorrect representation of the actual values present in the column. Hence, the corresponding columns were converted to numeric type which is either float or int depending upon the values present in the column for better interpretability using pandas. Additionally, the column which represented the date on which the observations were recorded was also of object type. This column was separately converted using the pandas function todatetime to manage and manipulate date and time seamlessly.

Machine learning models cannot interpret the labels or values of categorical columns directly. Hence, techniques such as one hot encoding and label encoding is used to create binary vectors of each input value, where only a single element of the vector is set to 1 and the rest are set to 0. This makes it easier for the model to understand the relationship between input values and the target variable. Categorical columns can either be nominal or ordinal. A variable is said to be nominal if it represents distinct labels for different categories which have no particular order. For instance, there are two categories in a column: Dog and Cat. Here, we cannot say that the label dog is greater or less than cat and vice versa. Ordinal variables on the other hand represent distinct labels that have a meaningful order. For example, a column has three categories: Low, Medium, High. Here, we can say that low $<$ medium $<$ high. Since there were no ordinal columns in the dataset, one hot encoding was performed on categorical variables.

Issues with the numerical target variable

Initially, looking at the data one would find percentage bleaching column to be the ideal target variable for a regression task because it is numerical and the remaining independent variables are linked to it. But, from the distribution of this column as shown in figure 2.3 , it can be seen that the distribution is skewed towards the left. This indicates that most of the observations in this dataset had their bleaching percentages near zero. This skewed data can make it difficult for any regression model to interpret the relationships accurately which might result in the model favouring the occurrence of certain data points which in this would be the data points which have bleaching percentages near zero. These values cannot be treated as outliers and removed because that would lead to a loss of lot of data.

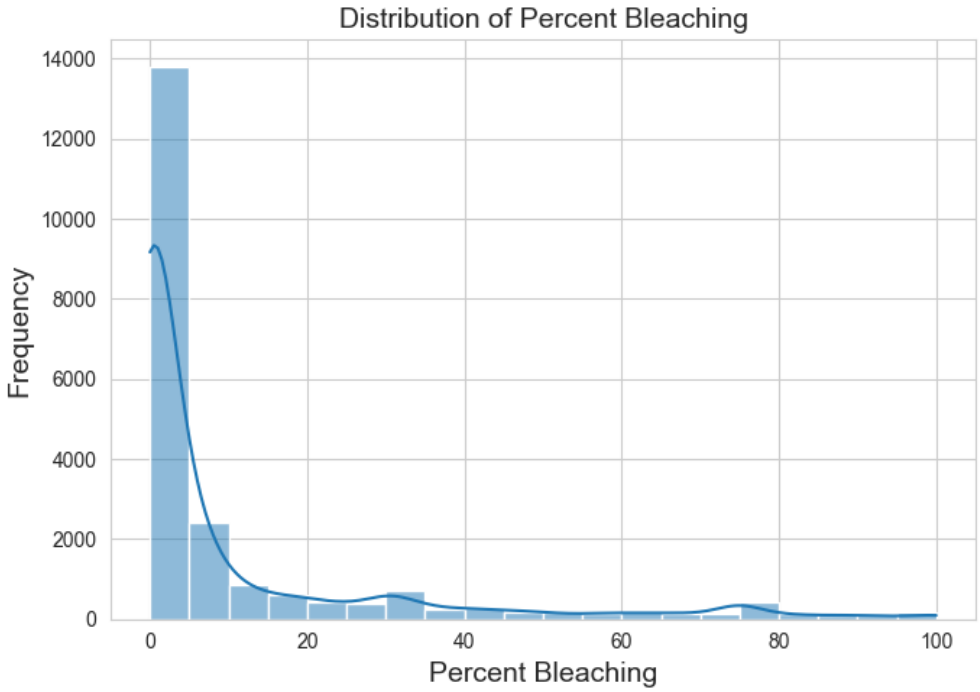


Figure 2.3: Distribution of 'Percent_Bleaching' column.

Creating binary target variable

The number of instances with value zero for bleaching percentage is 398490 which is way more than the instances for any other percentage. One possible solution to this issue is adding a new binary column named 'Bleached' using the existing column Percent Bleaching. Where, a value of 1 was assigned if the site had bleaching greater than 0. In this way, the distribution of both the classes 0 and 1 are close (count of the values). Now that the distribution is balanced as shown in figure 2.4, the model can effectively distinguish between the classes and eventually prevent the biasness towards any particular value which was the main issue when the target variable was numerical. Although this is a temporary fix. The working of regression models with the ideal target variable is explored in later sections

ENSO cycle events

The addition of the El-Nino and La-Nina into the dataset was an attempt to classify whether the recorded observation occurred during any of those cycles or not. First lets understand what these events are are how they can affect the corals.

An event is said to be of the EL-Nino criteria if the average sea surface temperatures in the Nino -3.4 region of the equatorial Pacific Ocean were at least 0.5 degree celsius warmer than average in the preceding month and has persisted or is expected to persist for 5 overlapping 3-month periods. Whereas, an event is said to be of the La Nina criteria if the average sea surface

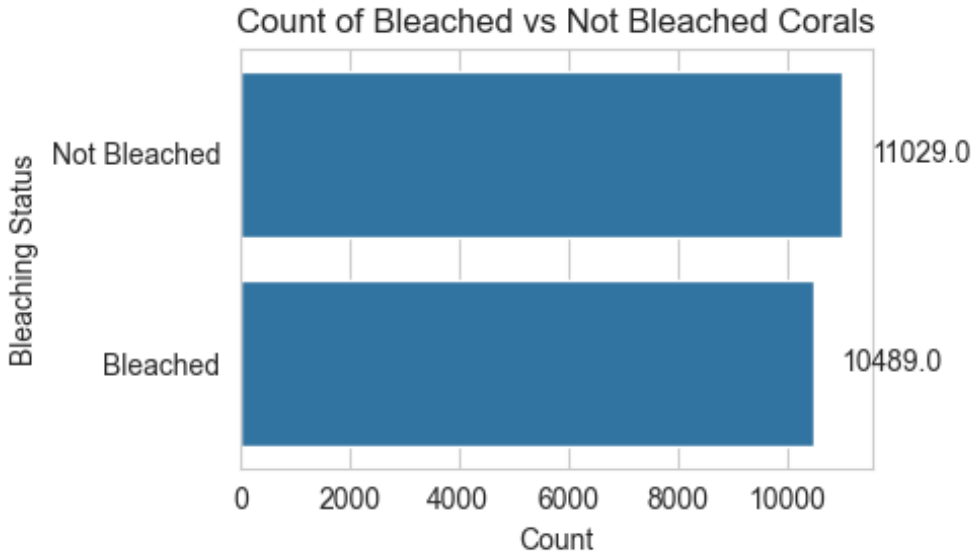


Figure 2.4: Distribution of Bleached vs Not-Bleached.

temperatures in the Nino-3-4 region of the equatorial Pacific Ocean were at least cooler than average in the preceding month and an average anomaly of at least -0.5 degrees has persisted or is expected to persist for 5 consecutive, overlapping 3-month periods. The anomalies are measured over three consecutive months which are represented as 'DJF' which is December, January, February. It is important to take into account the occurrence and impact of these events because according to research by [6] mass coral bleaching events occurred with increasing frequency since the early 1980s which often coincide with El-Nino events.

Unfortunately, from our dataset it is not possible to tell to which of the events a particular instance belongs to. Even though there is data from years 1980 - 2021, the data is not available for every month of each year. which is why it is not possible check for the events according to the consecutive three month period as mentioned above. So, to include this information to the dataset a different data was acquired from National Oceanic and Atmospheric Admistration (NOAA) which was initially in the form of an ascii text file. The file was converted to a CSV file so that it can be easily integrated along with our original dataset.

2.3.1 Adding cycle to data

In order to add the ENSO cycle information in the original dataset, we wrote a function that uses values from the ENSO cycle data and applies the results on the original set. The results of the function are added as values in the 'term-type' column representing whether an instance was in the neutral phase, El-Nino phase, or the La-Nina phase. As mentioned in the previous section, both El-Nino and La-Nina phases are checked through anomaly values (difference in sea surface temperatures compared to global average). For example, if the average difference

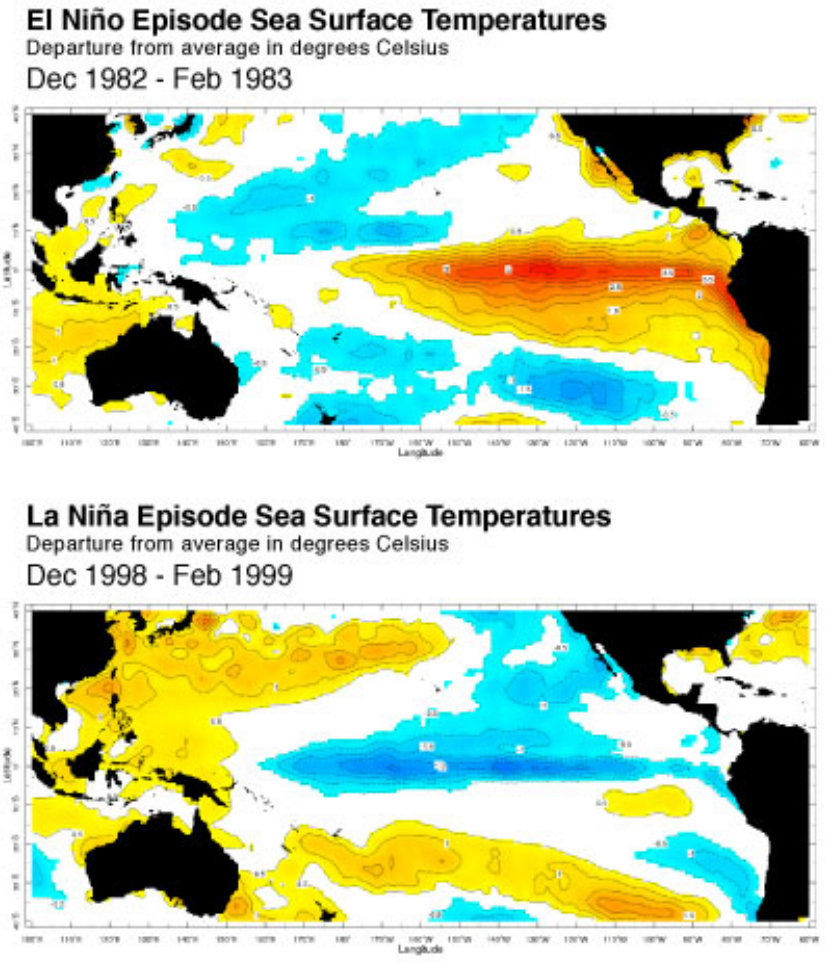


Figure 2.5: ENSO Phases [6]. The reddish-orange colour indicates the onset of warm temperatures whereas the dark blue colour indicates the onset of colder temperatures.

across a span of 5 consecutive 3 month periods is above 0.5 degrees then the instance from that particular month and year is classified as El-Nino for term-type.

Before we start describing the function, we start by importing the ENSO data and create a dictionary to add a numerical column for consecutive month entries. This means that a combination of 'DJF' is converted to '12-1-2' where the numbers represent the month in the order of the regular calendar. A new column is added to the ENSO data that contains only the numerical combinations of months. Another advantage of converting the months to numbers is that certain months have the same starting letter (March and May) which might result in confusion regarding the consecutive periods.

Now we try to understand how 5 consecutive 3 month periods can be identified from the dataset and then the goal will be to obtain averages of temperature anomaly values. For the explanation, we will use an example scenario where an instance in the original dataset has the

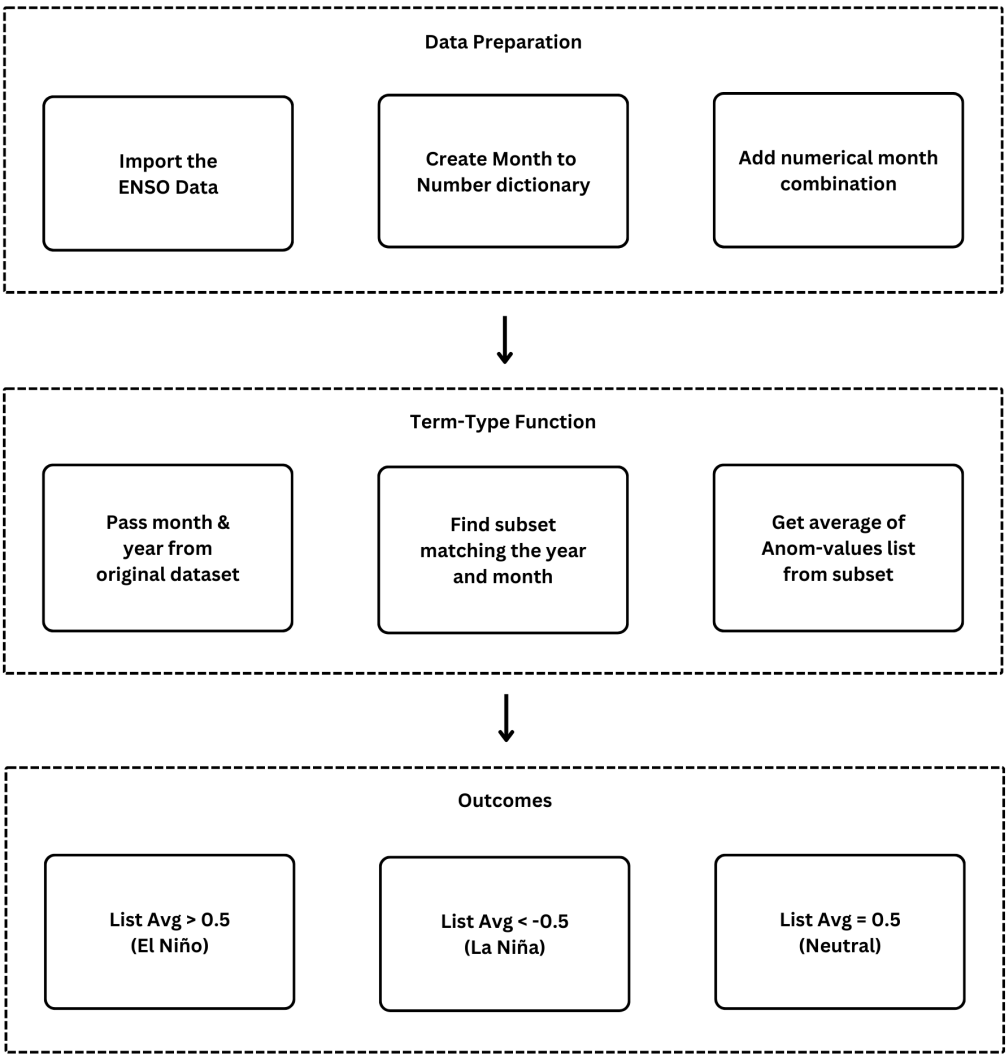


Figure 2.6: Creating the Term-Type function.

date of February 2010. In this case, the combinations with the month of February are– '12-1-2' (DJF), '1-2-3' (JFM), and '2-3-4' (FMA). These combinations can be identified by checking for the value '2' in the newly added column. This check would result in all rows where the month February is present. With this approach, we are able to find 3 consecutive 3 month periods but in order to find the other 2 combinations, we use the values before and after the matching combinations. In simple terms, rows pertaining to combinations '11-12-1' (NDJ) and '3-4-5' (MAM) are used to complete the 5 consecutive 3 month period that is needed to check for the ENSO phases. This approach was chosen such that the month from the original instance (February) stays in the middle of the period. Another important thing to note from the addition of new combinations is that the row indicating '11-12-1' (NDJ) represents the months before the original month of February. This means that in order to use the value from this combination, we will have to look for ENSO data from the previous year (2009).

On the flip side, if the example instance had the month of November, we would require the values of 3 month combinations from the next year (2011). This issue was tackled by using a split approach for different months in the year. If the month from the original data falls between January and February, then we would need the data from the previous year. But if the month fell between November and December, then the next year's data is accessed. Any month from the middle of the year only needs the ENSO data from that particular year. For example, the month August will have the set of– (MJJ), (JJA), (JAS), (ASO), (SON).

Once the period of 5 consecutive 3 months is identified for an instance, we access the anomaly values from the ENSO data and add them to a list. Then, an average of the 5 values from the list is obtained. The last part of the function is to return the respective ENSO phase that the month and year combination from original data represent. If the average turns out to be equal to or more than 0.5 degrees, then the function returns 'El-Nina' as a string. On the other hand, the string 'La-Nina' is returned if the average is less than or equal to -0.5 degrees. In the case where the average does not fall into either of the ranges, the string 'Neutral' is returned.

Now that the function is written, we use the 'apply' method with the 'lambda' function to find the term-types of each instance in the original dataset. The returned values are added as the 'term-type' column in the original data. After running the function on every instance, we find that almost 57% of the values (18,597) were recorded during the Neutral phase. A fraction of 24% of the instances (7,885) were in the La-Nina phase and almost 19% of them (6,196) were in the El-Nino phase. This ratios of all three term-types correlate with the fact that the El-Nino and La-Nina phases occur in spans of 2-7 years. While it is also important to note that the count of instances across the years (and months in an year) have been different throughout the dataset. This means that recent years with more recorded instances might be better indicators for term-type. Whereas, instances that are the only ones from a particular year might fail to show a clear pattern of ENSO cycle occurrences. With the addition of the term-type column, this information regarding ENSO phase can be passed to the models along with the recorded environmental factors.

Chapter 3

Modelling

Now that we have a better understanding of the dataset with the addition of climatic events such as El-Nino and La-Nina, we will begin to explore the possibility of capturing the impact of environmental factors on coral bleaching. With the use of machine learning models, we will try to address the task of classifying which corals have been bleached and also try to predict the amount of corals that have bleached at different sites. Since the dataset used consists of instances from different oceans, we will begin the classification with the assumption that across different sites the correlation of environmental factors with bleaching remains fairly consistent. In case our classification models produce results that show that a relationship between the independent variables and the target variable has been established, we will be able to predict whether new instances of site recordings have been bleached. On the other hand, an extension to the predictions would be to predict the percentage of bleaching. Through this section, we test multiple classification and regression models in an attempt to perform both the tasks with the best possible accuracy.

3.1 Classification models

In our first modelling task, we will aim to predict which of the many corals have bleached. For all classification models, the target variable is set to be the 'Bleached' column which is a binary categorical variable with values of either 0 (Not Bleached) or 1 (Bleached). The following section describes 4 different classification models that can be used for this task. The models are– Logistic Regression, Decision Trees, Random Forests and K-Nearest Neighbours.

In the previous chapter, we prepared our dataset for modelling. This processed dataset will be used for all the models ahead. Since all instances in the dataset contain true values for the 'Bleached' column, we can split the dataset into subsets for training and testing the models. This split is performed using Scikit-Learn's 'train_test_split' function that allows us to pick the ratio of the split and set other important parameters. With the availability of more than 21 thousand instances split in almost a similar count of bleached and non-bleached corals, the ratio of 80:20

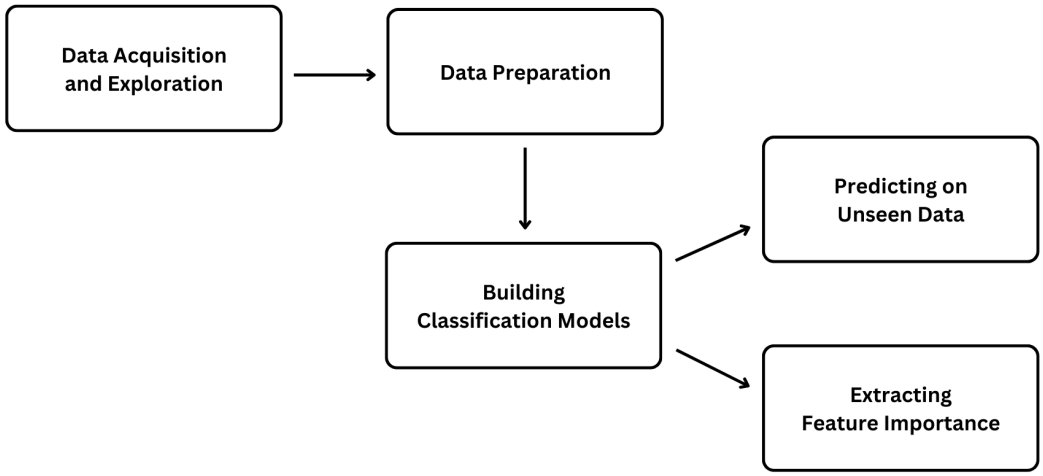


Figure 3.1: Workflow of classification model.

was chosen. This means that 80 percentage of the processed dataset will be used to train the models and the rest (20 percentage) will provide unseen instances to test the performance of our models. Testing on unseen data provides a check on overfitting which is the case where machine learning models learn a bit too much from the training instances and fail to apply the learning on unseen instances. If this is not checked and prevented, the model will initially show very good results while training but even slight changes during testing can impact performance.

Another parameter in the split function is 'random_state' which specifies a constant pattern of randomization when building train and test sets. In simple terms, assigning a random state ensures that on each run the train and test sets would remain identical. This helps in comparing different models over the same set of instances. Additionally, the stratify parameter was also used while making the split to ensure that the distribution of classes is in accordance to the original dataset. Setting stratify=True will preserve this proportion in both the training and test sets while ensuring that the models do not become biased towards a certain class.

3.1.1 Logistic Regression

Logistic Regression is the first model that comes to mind when approaching a classification problem. The base of this method is the logistic function that is a sigmoid function as mentioned in 3.1, maps a weighted linear combination of features into real values between 0 and 1. These values represent the prediction of probability of belonging to a certain class of data [18]. If the probability estimated by the model is more than 50 percent then the model predicts that the instance belongs to the positive class which is '1'(Bleached) is this case, or else it predicts that it does not which implies that it belongs to '0'(Not Bleached). This probability is given by 3.2.

The sigmoid function $\sigma(z)$ is given by:

$$\sigma(z) = \frac{1}{1 + e^{-z}} \tag{3.1}$$

where,

z is the linear combination of inputs ($z = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n$)

- β_0 is the intercept (bias term).
- $x_1, x_2, \ldots, x_n$ are the input features.
- $\beta_1, \beta_2, \ldots, \beta_n$ are the weights (coefficients) associated with each feature.

$$\begin{aligned} \text{Probability of the positive class} &: P(y = 1) = \sigma(z) \\ \text{Probability of the negative class} &: P(y = 0) = 1 - \sigma(z) \end{aligned} \tag{3.2}$$

This model assumes that the instances are independent of each other, there is little to no multicollinearity i.e two or more features are not highly correlated with each other, there are no outliers present in the dataset, the dataset is large and that the independent variables are linearly related to the log odds of the dependent variable. The model may perform poorly if these assumptions are not actually true. For instance, having high multicollinearity can make it difficult for the model to estimate the coefficients efficiently and determine the actual effect of each variable which is likely in our case since the dataset has a higher dimension. Similarly, the presence of outliers can also lead to inaccurate predictions and unstable results. In this dataset there are extreme values but if those are dropped, it will lead to a loss of lot of instances which will make the dataset smaller. This might make the model prone to underfitting.

For this particular dataset, I used logistic regression because it is easy to interpret and it is a simple model. So in order to understand if other complex models like random forests are performing better it was important to understand the performance of simple models too.

3.1.2 Decision Trees

A decision tree creates an ordered sequence of decision points based on the dataset. As the name suggests, it is a tree that is grown by recursively creating one split after another, based on the values of input variables, so as to maximise the probability that we classify the target variable correctly. In simple terms, they use the logic of if-else statements to partition the data until the decision reached has a higher probability of being right. The tree starts from the entire dataset which is the root node and then it looks for the most accurate question or feature that can effectively divide the data into different classes. For instance, for iris flower classification, there are features like petal length, petal width, sepal length and sepal width. The decision tree will first choose a feature that will help create a best split of the dataset. To determine the best feature, it uses node entropy. Entropy as shown in 3.3 in general is a measure of uncertainity.

In the context of decision tree we can say that if a node has low entropy it is more predictable and if a node has high entropy it is less predictable. So, a node where all the data are part of the same class has zero entropy and a node where data are evenly split between two classes has entropy of 1. To decide on the best split, weighted entropy is used. The lower the loss of the split the better the split is. This loss is calculated using the equation 3.4

Let p_c be the proportion of data points in a node with label c. For p_c the entropy S is given by:

$$S = -\sum_{c} p_c \log_2(p_c) \tag{3.3}$$

If a given split results in two nodes X and Y with $N1$ and $N2$ total samples in each respectively. The loss (L) of that split is:

$$L = \frac{N_1 \cdot S(X) + N_2 \cdot S(Y)}{N_1 + N_2} \tag{3.4}$$

where, $S(X)$ and $S(Y)$ are entropies of node X and Y respectively.

In simple terms, the working of a decision tree algorithm is as follows:

1. The tree begins at the root node with the whole dataset.
2. The below mentioned steps are repeated until every node is either pure (contains only one class) or unsplittable (node having duplicate data):
    - Select best feature x and split value beta such that the loss of the resulting split is minimized.
    - Split data into two nodes, one where $x < \beta$, and one where $x \geq \beta$.

However, decision tree models are very unstable because slightly different data sets can produce very different results.

3.1.3 K Nearest Neighbours

K Nearest Neighbours is an instance-based classification algorithm. It works on the principle of distance. When the algorithm encounters an unseen data point, it decides the label of that data point based on the label of K of it is neighbours. Here, K represents the number of neighbours the algorithm will be looking at. If majority of the K neighbours belong to a particular class, then the unseen data point will also be classified into that class as shown in figure 3.2. Let us understand its working step by step.

1. The training and test data.
2. Set the value of K.

3. The algorithm then calculates the distance between each instance of the training data and the instance of the test data using distance metrics like euclidean, manhanttan, hamming etc.
4. The distances are sorted and stored in ascending order.
5. The K closest points are identified.
6. To classify the data point, a majority vote of labels of the K nearest neighbours is taken and the label that occurs most frequently is assigned [28].

The formula to calculate Euclidean distance is:

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{3.5}$$

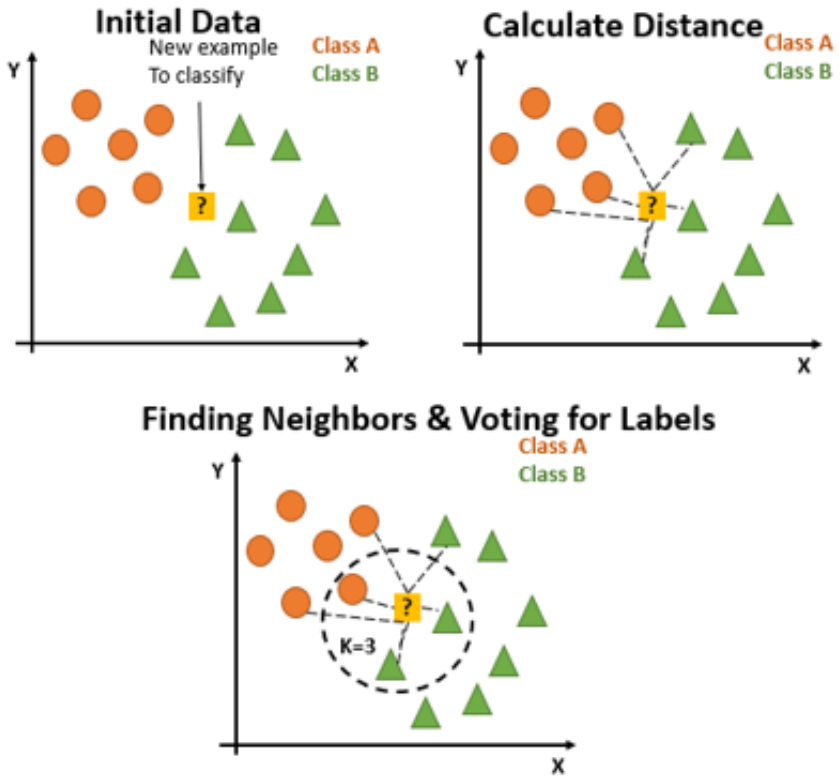


Figure 3.2: Example for KNN [33]. It shows how a KNN classifier determines the class of a new data point based on majority voting.

The essence of this algorithm lies in an optimal K Value. A small K value can lead to overfitting and inaccurate predictions whereas a larger value of K can lead to underfitting hence, it is important to choose an ideal value of K. It is usually recommended to choose an odd value

of K. Although there are no pre-defined methods to determine the k value, it can be done by cross-validation. In this the train data is divided into K equal subsets which is usually 5. Here 4 subsets will be used to find the optimal value of K and one will be used to validate it. On each of the 4 subsets, the K nearest neighbours model will be run using a range of K values(1-10). For instance, $K = 2$, on this value 4 subsets will be trained and the performance of it will be checked on the validation set. An average accuracy will be calculated for all 5 runs. This process will be repeated for all the K values. After this, the average accuracies across all K values is compared to choose the k value with the best accuracy.

Even though this model is easy to implement and understand, it is computationally expensive since it calculates distance for each instance and it is highly sensitive to the choice of K value. Additionally, it is sensitive to outliers and higher dimensions.

3.1.4 Random Forest Classifier

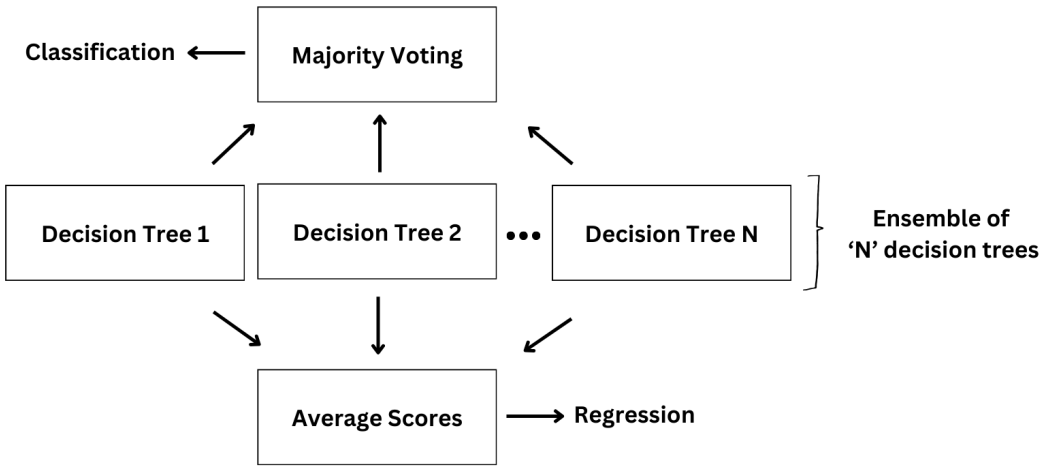


Figure 3.3: Random Forest

The Random forests classifier works on the principle of collective intelligence. According to Condorcet's Jury Theorem [10], a decision achieved by majority vote has a higher probability of being right rather than a group of independent voters. For instance, there are N individuals tasked to decide on the truth of some proposition. If each individual has a probability of being accurate p, and that $p > 1/2$ (One individual is more likely to be correct than wrong). If these N individuals vote, and the vote of individual i is X_i, with $X_i = 1$ implying the correct decision and $X_i = 0$ the wrong decision, then the probability that the group will make the correct decision is:

$$P(\text{Group is correct}) = P\left(\sum_{i=1}^{N} X_i > \frac{N}{2}\right) = P\left(Z > \frac{N}{2}\right), \quad Z = \sum_{i=1}^{N} X_i \sim B(N, p) \tag{3.6}$$

Similarly, if instead of N individuals there is a collection of classifiers. Even if on its own a classifier is showing poor performance, collectively their performance can be very accurate. Let us look at the random forest algorithm.

1. N bootstrap samples are created from the original data set.
2. One decision tree model is fitted on each bootstrap sample.
3. Each of the data point from the test data set is passed as input for each of the decision tree and predicted class labels are collected.
4. The final class is decided using majority voting.

Random forests apply an ensemble technique called bagging which is bootstrap aggregating. In this technique, multiple subsets of the same size are created from the training data are which are referred to as bootstrap samples. These samples are generated at random with replacement which means that the individual data points can be chosen more than once. Each of this subset is then used to train a decision tree independently. This ensures diversity in the samples, reducing overfitting. Due to this, Random forests are perfect for complex datasets with higher dimensionality like ours. Another advantage is that it does not require feature scaling.

3.2 Regression models

The next part of the modelling section would be to use the learning from classification and build regression models that can predict the percentage of bleaching on corals that were classified to be bleached. As previously discussed, the original column on bleaching percentage is severely skewed with most values equal or closer to zero. We will try to address this issue for our models to capture and reproduce the values on instances. Another important information obtained from classification models is the importance of features with respect to the target of bleaching percentage. This list of feature importance will help us on deciding the input features for regression. The idea here is to start with a baseline model and then improve the performance by tuning model parameters and applying transformations.

The processed data is split similarly as done in the classification section with a ratio of 80:20 and the same random state as before. Only difference here is that the target is now a continuous numerical column pertaining to percentage values. This part of modelling is not expected to provide ground breaking results due to the skewed target variable but with certain changes it can act as an extension to the classification task.

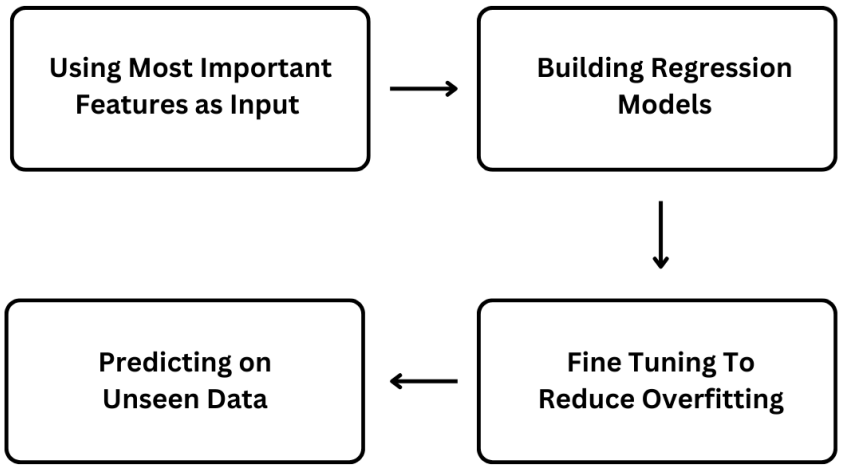


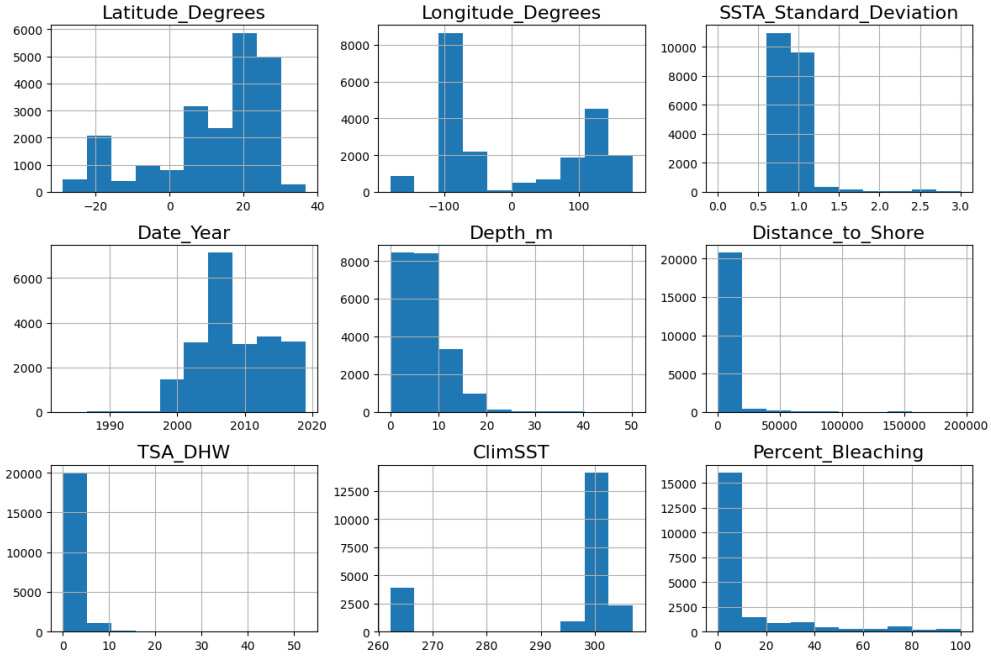
Figure 3.4: Workflow of regression model.

3.2.1 Log transformation

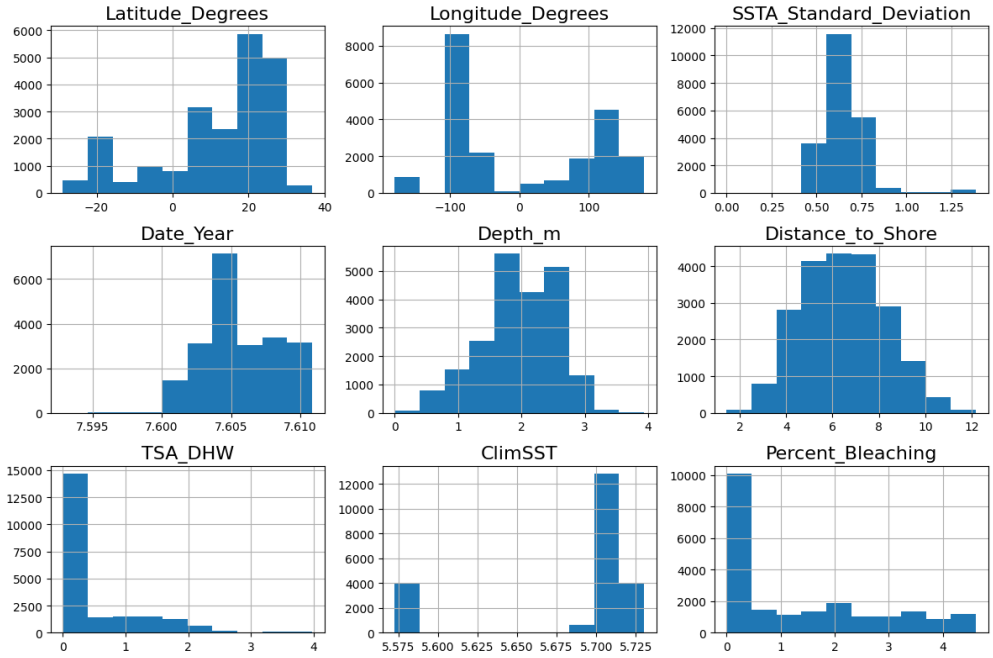
In the process of building the classification model few key issues were identified: Almost all of the variables do not have a symmetrical distribution which might be due to the presence of outliers. In general practice, the outliers could be removed to ensure better data quality. However, in this case, removing outliers will lead to a loss of a lot of the data which would not be ideal. Therefore, a better approach would be to transform the variables before passing them into the machine learning models to avoid misleading results.

In the process of data transformation, a mathematical operation is performed on each observation of the data set, then these transformed observations are used in the models. Described below are a few reasons to apply transformation [7]:

1. **Minimizing Skewness**: A distribution that is symmetric is easier to interpret rather than a skewed distribution. Furthermore, it will make the task of machine learning models uncomplicated while also reducing favouritism towards certain observations due to their scale or prevalence. It is a common practice to take roots or logarithms to reduce right skewness. Whereas to reduce left skewness, taking squares or cubes is preferred.
2. **Addressing Heteroscedasticity**: Heteroscedasticity refers to the state in which the variability of a variable is unequal across the range of values of a another variable that predicts it. This variance can be stabilized by applying transformations.
3. **Interpret linear relationships**: In cases like ours when the goal is to establish how bleaching percentages are influenced by parameters such as temperature, wind speed, and other parameters, it would be easier to interpret and analyze these relationships if the patterns in the plots are linear or indicate clear trends.



(a) Before Log Transformation



(b) After Log Transformation

Figure 3.5: Log transformation of features showing the distribution of each feature.

As we can see from the figure 3.5(a), the distribution of most of the selected features such as $Depth\_m$, $Distance\_to\_Shore$, TSA_DHW is also skewed including the target variable for regression model which is $Percent\_Bleaching$. After applying log transformation to the features, we can see from figure 3.5(b) that the features are now normally distributed which

is also indicated by the symmetrical arrangement of data points. Now, these log transformed features will be used as input features for the regression model to predict the target variable $Percent\_Bleaching$.

3.2.2 Linear Regression

A linear model makes a prediction by simply computing a weighted sum of the input features, plus a constant called the bias term also known as the intercept term as shown in equation 3.7.

$$y = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n \tag{3.7}$$

Where

- y is the predicted value
- n is the number of features
- x_i is the i^{th} feature value
- θ_j is the j^{th} model parameter (including the bias term θ_0 and the feature weights $\theta_1, \theta_2, \ldots, \theta_n$)

The model here tries to find the relationship between input variables which are independent variables and output variable which is the dependent variable. Before fitting a linear model we must check the correlation among the dependent variable and the independent variables. This model is sensitive to the presence of outliers.

3.2.3 XGBoost Regressor

It is also known as extreme gradient boosting is an extension of the gradient boosting algorithm which similar to random forests works on the principle of collective intelligence. To begin with, Let us understand gradient boosting first. Unlike random forests which utilise bootstrap aggregating also known as bagging, Gradient boosting algorithm as the name suggests uses boosting.

As discussed above, in bagging the for each bootstrap sample generated a new decision tree is trained. Although in boosting, models are trained sequentially. The aim is to correct the predictions made by the preceding model. Gradient boosting applies the core idea of boosting while minimizing errors using gradient descent. Here, each successive decision tree is trained on the residual errors made by the preceding decision tree. The difference between bagging and boosting is also illustrated in figure 3.6.

Key differences between Gradient boosting and extreme gradient boosting which makes XGBoost an ideal choice for our data set:

1. Gradient boosting is computationally expensive whereas XGBoost is not, making XG-Boosting ideal for a high dimensionality data set.

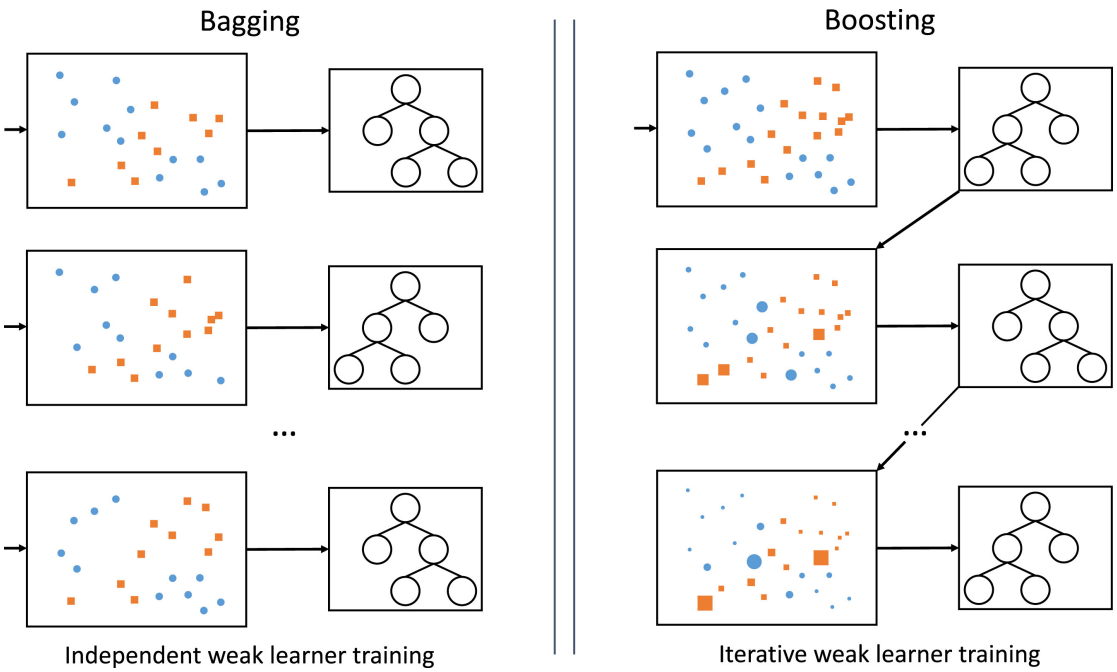


Figure 3.6: Bagging vs Boosting [17].

2. XGBoost algorithm includes built-in regularization making it more robust. It also has built-in cross validation such as early stopping, which will stop the training if no further improvement is observed. This eventually minimizes overfitting.

3. XGBoost has more number of hyperparameters which gives us the more control and flexibity to tune the model according to our requirements. The hyperparameters used will be detailed in later sections.

3.2.4 Random Forest Regressor

In contradiction to the random forest classifier where the final prediction of an unseen instance is decided by majority vote as described above, in random forest regressor the final prediction of the unseen instance is the average of the predictions made by independent decision tree models on the bootstrap samples. The rest of the working principle is same, only difference is that earlier the target variable was binary and here the target variable is numeric and continuous.

This model is found to be the best performing compared to the XGBoost model, though with a small margin. Since random forests do not have built-in methods to tune hyperparameters, we have used certain hyperparameter tuning methods in an attempt to further improve the performance of this model.

3.2.5 Cross Validation

Machine learning models have both parameters and hyperparameters and these terms are quite different from each other. From instance, in the linear regression equation mentioned above, θ_j is a model parameter. Parameters are variables that belong to the model itself and are learned during the process of training. However, hyperameters are set before the training process. Their role is to determine how and what a model can learn and how well the model will perform on unseen data [2].

In the process of cross validation, the data is split into k-folds (equal subsets of data). For each combination of hyperparameters, the model is trained on $k - 1$ folds and validated on the remaining fold. This process repeats k times with each fold acting as the validation set once. This means for each combination of hyperparameters, the model is trained and tested k number of times. To select the best set of hyperparameters, the combination that results in the highest average of specified evaluation metric (such as Mean Squared Error) across all the folds is chosen.

3.2.6 GridSearchCV

GridSearchCV is one of the commonly used hyperparameter tuning method from the Scikit-Learn library that helps in identifying the best combination of hyperparameters that can yield to improved performances for the data at hand. It uses the principle of cross validation, where it uses different subsets of the data to train the model multiple times over different combinations of hyperparameters. These combinations are obtained from a user-defined dictionary of hyperparameters and the lists of values that can be picked for each. This dictionary acts as the pool of available hyperparameters and is also called the 'grid' for the GridSearchCV. This grid can either be small or have an extensive list of options. For example, a small grid would have 2 hyperparameter options with 3 values for to select for each. But on the other hand, an example for a large grid can have 7 hyperparameters to chose from with a list of 10 values for each. GridSearchCV picks every single possible combination of hyperparameters and trains the model over the defined number of folds. Although this seems like the best way to find the best combination, this approach is often considered time and resource expensive as it performs an exhaustive search over the grid. Hence, it is advisable to use this approach when the grid has fewer options to train over or when there is an availability of resources to perform exhaustive searches [15, p. 79].

3.2.7 RandomizedSearchCV

RandomizedSearchCV module is also a part of the Scikit-Learn library. It is a technique used for hyperparameter tuning while also applying the principles of cross validation. Unlike GridSearchCV which attempts to find the best possible combination of hyperparameters through exhaustive search of all possible combinations, RandomizedSearchCV finds the best combina-

tion over a fixed number of parameter settings. In simple terms, this approach uses the grid mentioned in the previous section and randomly picks combinations instead of training models over every possible one. This solves the issue with GridSearchCV by reducing the time and resource complexity of the search but also comes with the trade-off that it does not necessarily identify the best combination each time. Since the process of picking combinations from the grid is randomized, there might be instances where this approach misses out on the best possible combination. But with this trade-off we can still obtain a general idea of how randomized combinations of hyperparameters and values can improve the model performance. This approach is used when there is a need for computational efficiency barring the slight decrease in the quality of the results [15, p. 81].

Chapter 4

Results

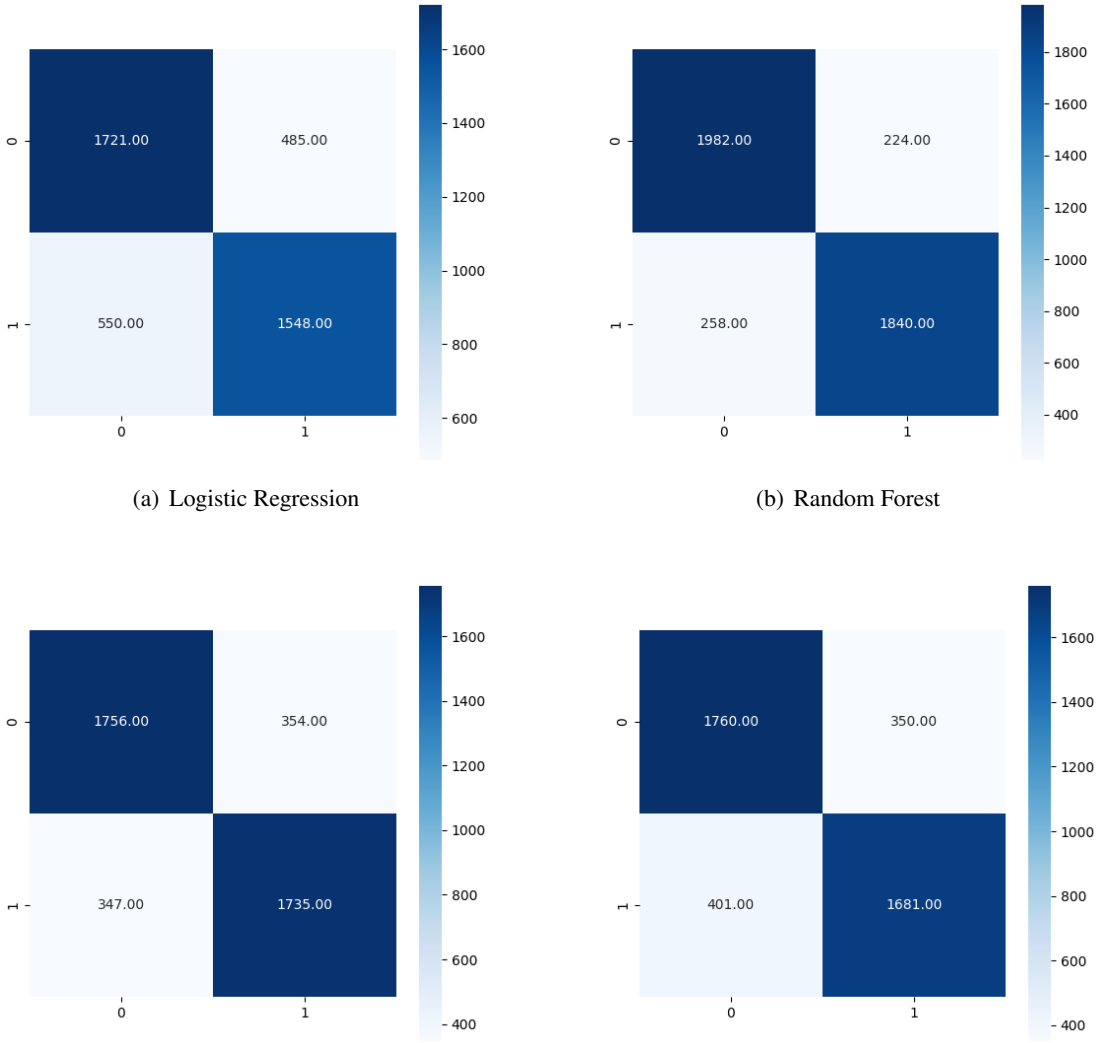
Let us look at the results of the classification model first. So, as mentioned above four classification models were run: Logistic Regression, Random Forest, Decision Tree and K nearest neighbours. Among these models the random forest classifier performed the best with the accuracy of 88.38 percent on the test data set. The aim of this classifier was to classify bleached corals instances as 0 and non-bleached coral instances as 1. Additionally, to identify the most important features that influenced the performance of the model the feature importances method was implemented which is a part of the random forest module in python. To evaluate the performance of classification model there are various known techniques such as confusion matrix, accuracy, ROC curve, precision, recall and F1-score. Our models will be checked on all of these metrics to find the model whose performance is above average in all or most of the aforementioned evaluation metrics. In this process, we check how well the models are able to predict the correct class labels which is 0 and 1 in our case (where 0 is not bleached and 1 is bleached).

4.1 Confusion Matrix

A confusion matrix is a table that categorises predictions according to whether they match the actual labels. It has four components: True Positive, True Negative, False Positive and False Negative. The class of interest is the positive class, while all others are negative class.

- True Positive(TP): Instances that are correctly classified as the positive class.
- True Negative(TN): Instances that are correctly classified as the negative class.
- False Positive(FP): Instances that are incorrectly classified as the positive class.
- False Negative(FN): Instances that are incorrectly classified as the negative class [15, p. 92].

From the confusion matrices shown in figure 4.1 for logistic regression, random forest, decision trees and K nearest neighbours. It can be seen that random forests have a better ability



(a) Logistic Regression

(b) Random Forest

(c) Decision Trees

(d) K Nearest Neighbours

Figure 4.1: Confusion matrices of classification models.

to capture and distinguish between the classes than logistic regression. An ideal classifier would have no false positives and false negatives. But since it is not possible to achieve that in real life, we go with the one which has fewer instances of false positives and false negatives which in this case is random forest with 224 and 258 instances of false positives and false negatives respectively. The the number of true positives and true negative instances are also more in random forests than in logistic regression.

4.1.1 Interpreting the model performance

From the table we can observe the values of train and test accuracies, precision, recall and F1-score. These values give an idea about the performance of the classification models numerically.

Model Name	Train Accuracy	Test Accuracy	Precision	Recall	F1-score
Logistic Regression	75.25	75.95	0.76	0.73	0.74
Random Forest	99.83	88.80	0.89	0.87	0.88
Decision Tree	99.84	84.97	0.83	0.85	0.84
K Nearest Neighbors	85.42	79.11	0.80	0.75	0.77

Table 4.1: Evaluation metrics for classification models.

The train and test accuracies represent the performance of the model on train and test data set respectively. It is important to assess the performance of the model on test set since that would be an accurate measure of how well the model does on previously unseen data. To interpret F1-score it is important to understand the following key terms:

1. Sensitivity: It is also called as Recall or True Positive Rate. This values tells how many of real positives were correctly predicted [15, p. 94].

$$\text{Sensitivity (Recall)} = \frac{TP}{TP + FN} \tag{4.1}$$

2. Specificity: It is also called as the True Negative Rate. This value tells how many real negatives were correctly classified [15, p. 93].

$$\text{Specificity} = \frac{TN}{TN + FP} \tag{4.2}$$

3. Precision: It aims to answer the question: "How many positive predictions are truly positive?". The difference between precision and recall is that precision focuses on prediction results such as how useful the prediction results are whereas recall focuses on samples such as how complete the predicted results are taking into account all true samples [15, p. 93].

$$\text{Precision} = \frac{TP}{TP + FP} \tag{4.3}$$

4. F1-score: It is a combined measure to assess a given model in terms of both precision and recall together [15, p. 95].

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4.4}$$

Ideally we would want precision and recall values to be one since only then there would be zero false negatives and false positives. But in reality, a good classifier would have the respective values close to one. In this case, the random forest classifier, outperforms other models in every evaluation metric with test accuracy of 88.80 percent, precision of 0.89, recall of 0.87 and f1-score of 0.88.

4.2 Interpreting the ROC curves

A Receiver Operating Curve (ROC) is a diagnostic curve which assesses the overall performance of any given model. It does so by plotting the true positive rate(y-axis) against the false positive rate(x-axis). True Positive Rate(TPR), also known as sensitivity, is the proportion of instances which are actually bleached and are classified correctly. The False Positive Rate also known as specificity, is the proportion of instances that are not bleached but are incorrectly classified as bleached. An ideal classification model, would have FPR of 0 and TPR of 1. The Area Under the Curve (AUC) represents the ability of the classifier to distinguish between the classes which means, higher the AUC the better the model's performance [21]. From the ROC curves shown in figure 4.2, it is evident that the random forest model has performed the best with an AUC value of 0.95.

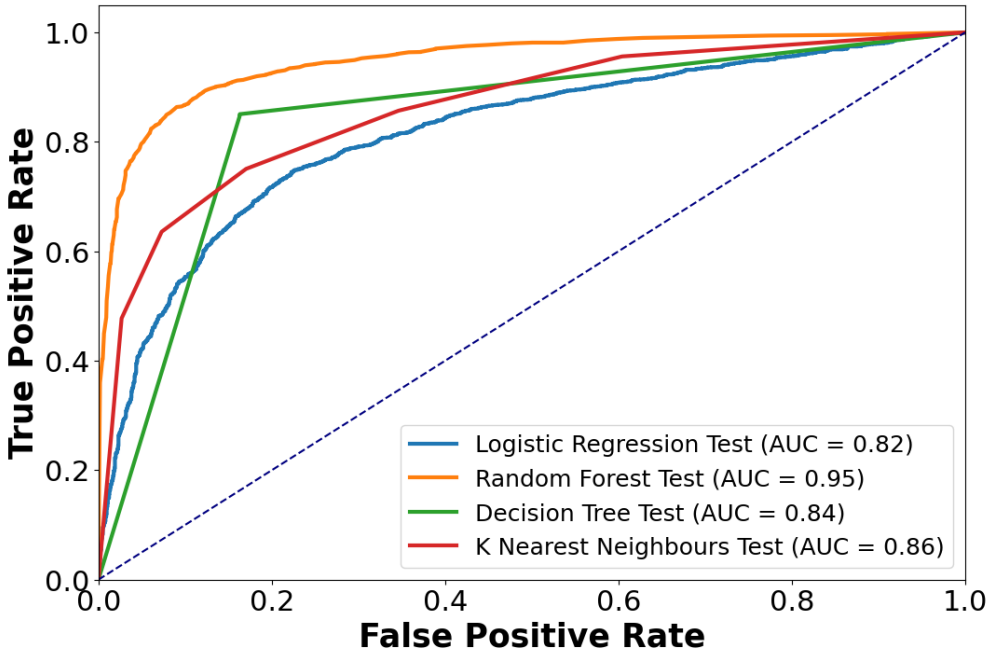


Figure 4.2: ROC curves for classification models.

4.3 Extracting feature importances

The most important feature of the decision tree and random forest classifier is that they provide access to a module called feature importances. It assigns a score to all input features based on how useful they are at predicting the target variable [15, p. 200]. For this particular dataset, feature importance values would be very useful since it has a high dimensionality. As discussed earlier, one of the reasons to build classification model was to determine the most important features. Selecting only the most important features and then training the model will improve

the efficiency of the predictive models. As we can see from the plot of the feature importances in figure 4.3, we are able to identify which features are least relevant to the model. The feature selection was done based on the threashold value of 0.02, which means the features with a score of 0.02 or more were chosen to further check the performance of the best classification model and also in the regression models.

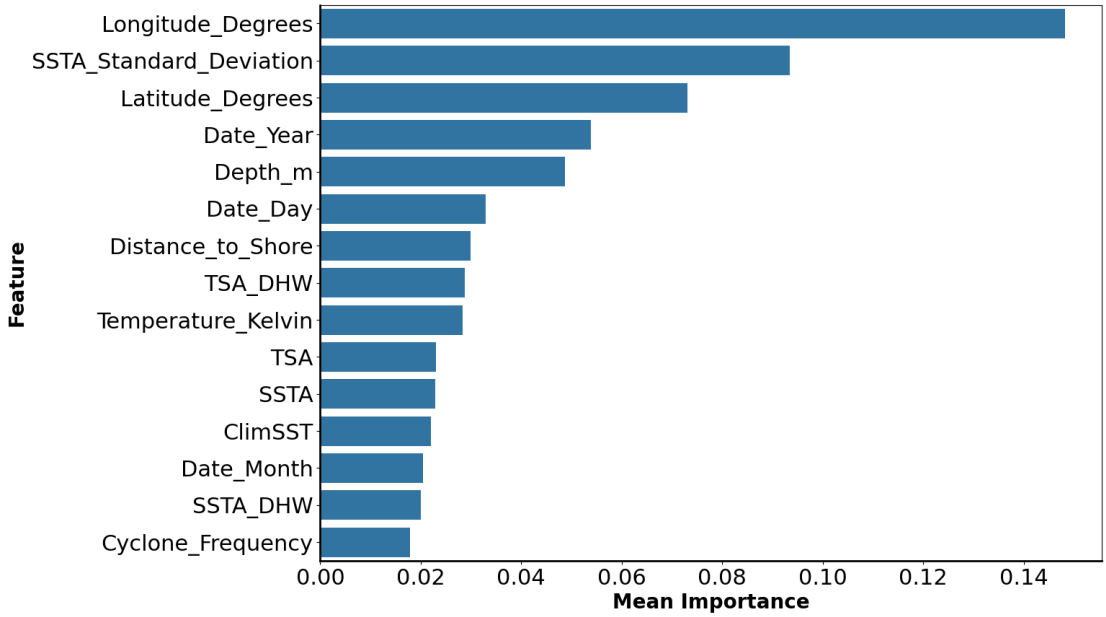


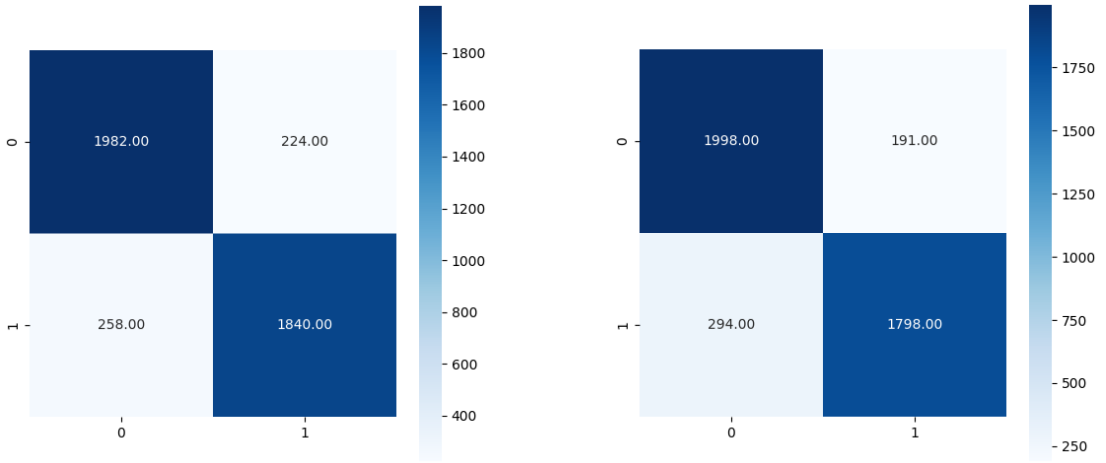
Figure 4.3: Feature importances plot using random forest.

To avoid repetitive features and multicollinearity, a few features were not used. The final features used are: $Longitude\_Degrees$, $SSTA\_Standard\_Deviation$, $Latitude\_Degrees$, $Depth\_m$, $Date\_Year$, $Distance\_to\_Shore$, TSA_DHW, $ClimSST$,$Percent\_Bleaching$.

The feature Date_Day was dropped since there was already the Date_Year feature which has a higher importance score while also specifying the year of each observation. Additionally, $ClimSST$ along with other temperature variables which were in degree Celsius. Hence, to maintain uniformity among features it was opted to disregard $Temperature\_Kelvin$ because it represents the value of $ClimSST$ which is the sea surface temperature in Kelvin.

4.3.1 Performance of random forest classifier after selecting the most relevant features

After selection of the most relevant features, our best performing classification model which is Random Forest was re-run and checked if there were any improvements in its ability to capture whether an observation belongs to the not bleached class or the bleached class. The use of confusion matrices to compare the performance provides an easy to interpret method of assessment where the amount of correct and incorrect classifications can be checked for both models.



(a) Before selecting important features

(b) After selecting important features

Figure 4.4: Confusion Matrix of Random forest classifier.

As can be seen from the confusion matrices above, the darker squares representing true values that were correctly classified have slight changes in value. Similarly, there does not seem to be a significant improvement in the negative classes (values that were classified in the wrong group). This result shows that using a selected set of features does not directly influence the performance of our classification model. On the other hand, this can also mean that the set of chosen features represent the predictive power of the model using all the features. This tells us that the set of chosen instances can be used ahead with the reason of logic, relevance in the context of coral bleaching, and results depicting comparable performances with or without the entire set of features.

4.4 Performance of regression models

Now that we are aware of the most relevant features that are responsible for bleaching. We can attempt to identify the exact percentage of bleaching of any given site. These features were used to train three regression models: Linear Regression, Random Forest Regressor and XGBoost Regressor and its performance was evaluated using metrics like Mean Squared Error(MSE), Root Mean Squared Error (RMSE) and R-squared value. The error term represents how the observed value is different from the actual value. Regression algorithms plot the best fit line that minimizes the error term. But not all points lie exactly on the line. Hence, the distance between each data point and the best fit line makes up the error term. Often residuals and error term are used synonmously, there is a very slight difference between the two[13].Residuals demonstrate how the predicted values differ from the actual values while the error term tells how far is the predicted value from the actual value.

- Mean Squared Error (MSE): The MSE value is calculated as the average of squared differ-

ence between predicted value and actual value of the target variable. This value is always non-negative; the closer the value is to zero, the better the performance of the regression model. This is because squaring penalises models for larger error values. For our regression models the MSE value was calculated using the mean_squared_error module from the scikit-learn library which applies the principle of equation 4.5.

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{4.5}$$

Where:

- n is the number of instances
- y_i is the actual value
- $\hat{y}_i$ is the predicted value.

- Root Mean Squared Error (RMSE): It is the square root of MSE value. It estimates the standard deviation of residuals. It is calculated using equation 4.6.

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{4.6}$$

- R-squared value: R-squared is a measure that rewards you for including too many independent variables, and it does not provide any incentive to stop adding more variables. Its value increases every time an independent variable is added to the model. Hence, when a model contains a lot of variables, it tends to capture every noise or random points in the data. This can lead to overfitting which is the model failing to generalise on the whole dataset. This issue is overcome by the adjusted R-squared value which increases only if a new variable improves the model fit [14].

4.4.1 Regression model

Initially, the regression models produced poor results. The linear regression model was the weakest model in our case. This could be due to the fact that even though there are relationships between the target variable and independent variables, it is too complex for the linear regression model to understand. Hence, it was opted to re-run the data on Random Forest regressor and XGBoost Regressor. Even though both these models yielded a very good accuracy on the training data, they failed to generalise the predictions on the unseen test data set which means that they were overfitting. To reduce this overfitting, on both these models hyperparameter tuning was performed using RandomizedGridSearchCV to correctly identify which of the models is the best predictor of the target variable without any overfitting.

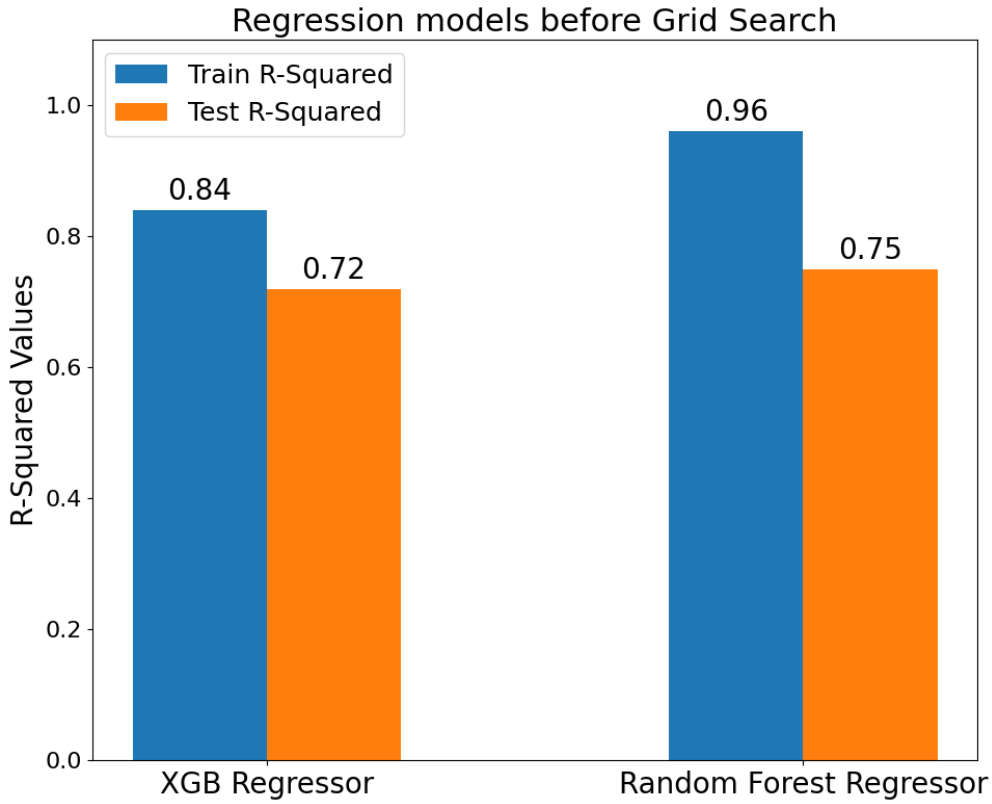


Figure 4.5: R-Squared values of regression models before hyperparameter tuning using Grid Search.

Hyperparameter	Overfit model	generalised model
bootstrap	False	False
max_depth	18	10
max_features	log2	log2
min_samples_split	4	2
min_samples_leaf	2	1

Table 4.2: Hyperparameters for Random Forest Regressor

The hyperparameters used in random forest regressor were learning_rate, max_depth,

max_features, min_samples_split, and min_samples_leaf. max_depth denotes the maximum depth of the tree which is the path between root node (starting node) and leaf node (last node with no further split). max_features is the number of features to take into account when finding the best split. min_samples_split is the minimum number of samples needed to split an internal node. min_samples_leaf is the minimum number of samples required to be at a leaf node. If bootstrap samples are used to build trees it is true, when it is false the whole dataset is used to build each tree.

The hyperparameters used in XGBoost regressor were learning_rate, max_depth,

min_child_weight, colsample_bytree, and gamma. Here max_depth means the same as the max_depth of random forest regressor. min_child_weight is the minimum sum of instance weight

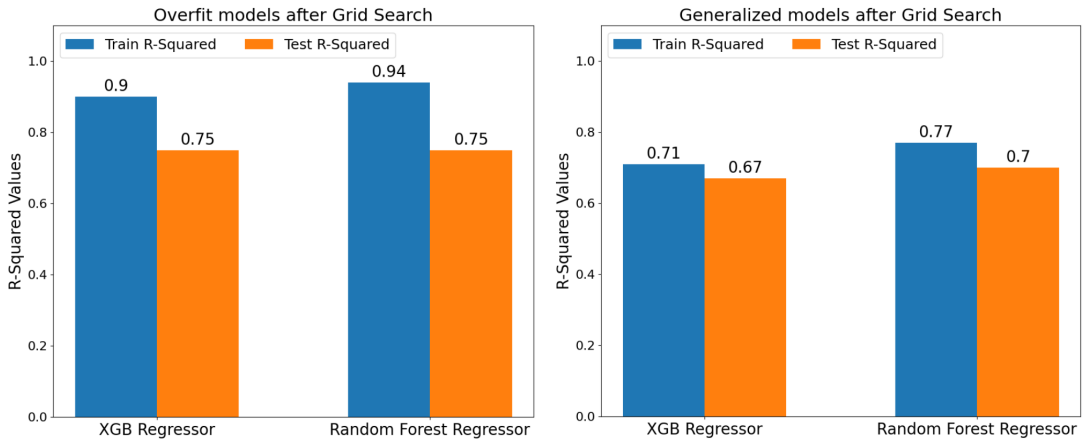


Figure 4.6: R-Squared values of regression models after hyperparameter tuning using Grid Search.

Hyperparameter	Overfit model	generalised model
learning_rate	0.05	0.05
max_depth	12	6
min_child_weight	5	3
gamma	0.0	0.0
colsample_bytree	0.5	0.5

Table 4.3: Hyperparameters for XGBoost Regressor

needed in a child. The lesser this value, lesser the overfitting. colsample_bytree is the subsample ratio of columns when constructing each tree. Minimum loss reduction necessary to make another partition on a leaf node of the tree is denoted by gamma. The learning_rate controls how each new tree contributes to the ensemble prediction. Lower the learning rate, the lesser is the model prone to overfitting.

Before performing GridSearchCV, there was a huge gap between the regressor model's performance on the train dataset and test dataset as shown in figure 4.5. Even though both random forest and XGBoost performed well on the train set, they failed to generalise on the test set, leading to a poor performance on the test set. This indicates that both the regression models are overfitting. Hence, to find which model is the best between these two we need to carry out GridSearchCV to simultaeneously reduce overfitting and choose the best hyperparameters. However, even after performing GridSearchCV on random forest and XGBoost regressors, there was little effect on the R-sqaured values on the test set as indicated in figure 4.6. The reason behind this was a higher value of the hyperparameter max_depth. On reducing the value for max_depth, it was observed that the model could generalise well on unseen data. But this was achieved by compromising on the performance of the model on the train dataset. Ideally, we should always prefer a model that generalises well on unseen data. In this way, the model will not be sensitive to the variance among the input parameters. In this case, the XGBoost regressor

performs better than random forest regressor in generalising on unseen data with a very small margin.

Future Work

During the course of this dissertation, with the aim of identifying coral reefs with high risk of bleaching we first tested multiple classification models. The results from the models showed decent performance with good generalisation. Followed by assessing the classification methods, a subset of climatic factors (inputs) was identified based on importance, logic, and relevance with the target of coral bleaching. To start building and testing models, few steps from the data science life-cycle had to be taken, namely– data cleaning, data pre-processing, modelling, and assessing results. All the functions related to the steps mentioned can be put in a common data preparation and modelling pipeline in the future. The pipeline will not only ensure uniformity in data structure but also be applicable on future updated versions of the dataset.

Since the issue with the distribution of the numerical target (bleaching percentages) was identified, the primary focus for the dissertation was to convert these percentages to classes. As in this case, any value above 0 was classified as Not-Bleached but in the future, a more flexible grouping of classes can be performed (for example, high risk, safe, neutral). Because of these issues in the available dataset, regression models were an extension to the classification of bleached vs non-bleached sites, the model's performance could be improved further in the future to make it more reliable and stable on unseen data.

In case the regression model performs considerably well on unseen data, it would further expand the scope of the project. One example of future approaches would be the addition of time series models to predict estimated bleaching percentages for sites in the future. With an analysis of past trends and future predictions, decision makers would be able to formulate better strategies to effectively manage and mitigate the damage.

In addition to modelling, an in-depth analysis of coral bleaching could be done with respect to a particular ocean, eco-region, country or city. This would be helpful to understand the patterns in the specified region of study. The main issue that was identified with the dataset is that there were a lot of missing values along with skewness. To address this issue in a better way, there could be separate datasets for observations with similar percentages of bleaching so that they can be analysed in a better way. Since the dimensionality of the dataset is high, we could also explore ways to effectively combine the information of two or more columns so we might not have to discard a lot of columns for analysis while also avoiding multicollinearity.

Conclusion

This project was built with an aim to understand and interpret the influence of climatic factors on coral bleaching. It was built around coral bleaching because mass bleaching event is occurring as this thesis is being documented and it is important to acknowledge the contribution of coral reefs not just in tourism industry but also in other aspects such as preventing erosion, controlling damage from waves and exploring the impact of climate change. With the climate getting increasingly unpredictable, action is necessary to safeguard coral reefs. To proceed with the analysis, the dataset was acquired from Biological and ocean Data management office. Along with the issue of missing values, an important problem was addressed with the help of binary target which was the skewness of the Percent_Bleaching column.

Classification models such as Logistic Regression, K Nearest Neighbours, Decision Trees and Random Forests were trained and tested for ability to rightly classify coral instances. Among the models, Random Forests performed the best displaying good performances on both the training and test sets proving itself to be a reliable model compared to others. Next, important features from the feature importances module of the random forest model were select to build the regression models. The most important features with mean importance of 0.02 and greater were 'Longitude_Degrees', 'SSTA_Standard_Deviation', 'Latitude_Degrees', 'Date_Year', 'Depth_m', 'Distance_to_Shore', 'TSA_DHW', 'Temperature_Kelvin','SSTA', 'TSA' 'ClimSST' in decreasing order of their importance. This result aligns with previous research that highlights temperature fluctuations as a major factor responsible for stress in corals.

Regression models were built in an attempt to predict the actual percentage of bleaching at a given site. However, The regression model did not perform as well as the classification models even after significant fine tuning of hyperparameters along with log transformation of features to address any skewness exhibited. These tweaks made the models generalise on unseen data, but this was at the cost of overall predictive performance. This could be primarily because of the skewness in the dataset and, much more importantly, due to inconsistencies in data collection time frames. In conclusion, we able to successfully process the data at hand and build strong classification models while also providing an extension with regression techniques and the addition of ENSO cycle information. The thesis can act as a reference for future research in the field and an extended study of ENSO cycles in this context can help in understanding the phenomena of coral bleaching better.

Bibliography

[1] Global Coral Reef Alliance. *Coral Reef Bleaching and Sea Surface Temperature*. URL: https://globalcoral.org/_oldgcra.

[2] Christian Arnold et al. "The role of hyperparameters in machine learning models and how to tune them". In: *Political Science Research and Methods* (2024), pp. 1–8. DOI: 10.1017/psrm.2023.61.

[3] Great Barrier Reef Marine Park Authority. *Coral Bleaching Fact Sheet*. Great Barrier Reef Marine Park Authority, 2019.

[4] Nathaphon Boonnam et al. "Coral reef bleaching under climate change: Prediction modeling and machine learning". In: *Sustainability* 14.10 (2022), p. 6161.

[5] *Climate Indicators: Sea Surface Temperature*. URL: https://climate.copernicus.eu/climate-indicators/sea-surface-temperature.

[6] *Columbia Climate School: ENSO Essentials*. URL: https://iri.columbia.edu/our-expertise/climate/enso/enso-essentials.

[7] Nicholas J. Cox. *Transformations: an introduction*. 2007. URL: http://fmwww.bc.edu/repec/bocode/t/transint.html.

[8] Kay S Dao HN Vu HT and Sailley S. "Impact of Seawater Temperature on Coral Reefs in the Context of Climate Change. A Case Study of Cu Lao Cham – Hoi An Biosphere Reserve." In: *Frontiers in Marine Science* (2021). DOI: 10.3389/fmars.2021.704682.

[9] J M S Delevaux et al. "Scenario planning with linked land-sea models inform where forest conservation actions will promote coral reef resilience". In: *Sci. Rep.* 8.1 (Aug. 2018), p. 12465.

[10] Franz Dietrich and Kai Spiekermann. "Jury Theorems". In: *The Stanford Encyclopedia of Philosophy*. Ed. by Edward N. Zalta and Uri Nodelman. Spring 2023. Metaphysics Research Lab, Stanford University, 2023.

[11] Adele M. Dixon et al. "Coral Reef Exposure to Damaging Tropical Cyclone Waves in a Warming Climate". In: *Earth's Future* 10.8 (2022), e2021EF002600. DOI: https://doi.org/10.1029/2021EF002600.

[12] Fernandez Emily. “A Study on Global Reef Deterioration: Exploring Coral Bleaching”. MA thesis. Claremont Mckenna College, 2023.

[13] *Error Term: Definition and Examples — statisticshowto.com*. [Accessed 07-09-2024].

[14] Jim Frost. *How to Interpret Adjusted R-Squared and Predicted R-Squared in Regression Analysis*. 2017. URL: https://statisticsbyjim.com/regression/interpret-adjusted-r-squared-predicted-r-squared-regression/.

[15] Aurélien Géron. *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow*. O’Reilly Media, 2019. ISBN: 9781492032649.

[16] Dr. Mathilde Godefroid. *The fourth mass coral bleaching*. 8/04/2024. URL: https://www.mpg.de/21887931/coral-bleaching.

[17] Sergio González et al. “A practical tutorial on bagging and boosting based ensembles for machine learning: Algorithms, software tools, performance study, practical perspectives and opportunities”. In: *Information Fusion* 64 (2020), pp. 205–237. ISSN: 1566-2535. DOI: https://doi.org/10.1016/j.inffus.2020.07.007.

[18] Harris JK. “Primer on binary logistic regression.” In: *Fam Med Com Health* (2021). DOI: 10.1136/fmch-2021-001290.

[19] Caroline Costa Lucas et al. “Turbidity buffers coral bleaching under extreme wind and rainfall conditions”. In: *Marine Environmental Research* 192 (2023), p. 106215. ISSN: 0141-1136. DOI: https://doi.org/10.1016/j.marenvres.2023.106215.

[20] Indeever Madireddy, Rachel Bosch, and Serena Mccalla. “Using machine learning to develop a global coral bleaching predictor”. In: *Journal of Emerging Investigators* (2023). DOI: 10.59720/22-056.

[21] Francis Sahngun. Nahm. “Receiver operating characteristic curve: overview and practical use for clinicians.” In: *Korean journal of anesthesiology* 75 (2022). DOI: 10.4097/kja.21209.

[22] NOAA. *Why are coral reefs important? National Ocean Service website*. 8/12/24. URL: https://oceanservice.noaa.gov/education/tutorial_corals/coral07_importance.html.

[23] NASA Earth Observatory. *Sea Surface Temperature Anomaly*. URL: https://earthobservatory.nasa.gov/global-maps/AMSRE_SSTAn_M#:~:text=Sea%20surface%20temperature%20is%20the,month%20from%201985%20through%201997.

[24] Deron Burkepile Robert van Woesik. *Bleaching and environmental data for global coral reef sites from 1980-2020*. 2022-10-14. DOI: 10.26008/1912/bco-dmo.773466.2.

[25] Tom Shlesinger and Robert van Woesik. “Oceanic differences in coral-bleaching responses to marine heatwaves”. en. In: *Sci. Total Environ.* 871.162113 (May 2023), p. 162113.

[26] Danielle L Spring and Gareth J Williams. “Influence of upwelling on coral reef benthic communities: a systematic review and meta-analysis”. In: *Proc. Biol. Sci.* 290.1995 (Mar. 2023), p. 20230023.

[27] P. K. Swart. *Coral Reefs: Canaries of the Sea, Rainforests of the Oceans*. 2013. URL: https://www.nature.com/scitable/knowledge/library/coral-reefs-canaries-of-the-sea-rainforests-97879685/.

[28] Kashvi Taunk et al. “A Brief Review of Nearest Neighbor Algorithm for Learning and Classification”. In: *2019 International Conference on Intelligent Computing and Control Systems (ICCS)*. 2019, pp. 1255–1260. DOI: 10.1109/ICCS45141.2019.9065747.

[29] *These photos show what happens to coral reefs in a warming world — nationalgeographic.com*. [Accessed 08-09-2024]. URL: https://www.nationalgeographic.com/environment/article/before-and-after-bleaching-lord-howe-coral-reef.

[30] Banyatsang Mphago Oteng Tabona Tlamelo Emmanuel Thabiso Maupong. “A survey on missing data in machine learning”. In: *Journal of Big Data* (2021). DOI: https://doi.org/10.1186/s40537-021-00516-9.

[31] H Triwibowo et al. “Python Programming for Degree Heating Weeks Estimation using Sea Surface Temperature Data from The Google Earth Engine Dataset as Coral Bleaching Analysis Tools”. In: *IOP Conference Series: Earth and Environmental Science* 1350.1 (June 2024), p. 012035. DOI: 10.1088/1755-1315/1350/1/012035. URL: https://dx.doi.org/10.1088/1755-1315/1350/1/012035.

[32] Robert van Woesik and Chelsey Kratochwill. “A global coral-bleaching database, 1980-2020.” In: *Scientific Data* (2022). DOI: 10.1038/s41597-022-01121-y.

[33] Lili Zhu and P. Spachos. *Support Vector Machine and YOLO for a Mobile Food Grading System*. Jan. 2021.