university of groningen / faculty of science and engineering / bernoulli institute

FishAI: Predicting high-likelihood Atlantic cod locations utilizing an ensemble Lasso regression model on geospatial data

Bachelor's Project Thesis

Lieke Bügel, s4341724, l.a.bugel@student.rug.nl
Supervisors: J.D. Cardenas Cartagena

Abstract: This study evaluates the performance of an ensemble Lasso regression model in predicting high-likelihood Atlantic cod locations and corresponding catch yield within the Norwegian Exclusive Economic Zone using historical catch data and environmental data. An ensemble Lasso regression model, consisting of multiple meta-models, was selected for this study, with its architecture providing both temporal and spatial context to the predictions. A sliding window technique generates input-output datasets, enabling supervised learning methods to capture temporal dependencies from the data. Each window consists of 5 input days of historical catch and environmental data, along with 5 output days of cod catch locations and associated product weights, with the model learning from these input-output pairs for future predictions. While the model successfully identified general trends in daily cod distributions, it showed signs of underfitting, as indicated by the simplified S-shaped predicted patterns and low R^2 scores. The mean squared error values indicate relatively low prediction error, but denormalized location and weight deviations highlight uncertainties about the model's practical utility. A critical limitation was the lack of spatial and temporal context in the input data.

1 Introduction

1.1 Motivation

Sustainable fishing is a growing concern, especially in Norway, where the fishing industry is the largest in Europe in terms of volume and value, significantly contributing to the country's GDP (Nordmo, Kvalsvik, Kvalsund, Hansen, Halvorsen, Hicks, Johansen, Johansen, and Riegler, 2022; Council, 2021). The Norwegian Exclusive Economic Zone (EEZ), which serves as its primary fishing zone, encompasses an area of approximately 2.1 million square kilometers of water (Nordmo et al., 2022).

The Norwegian fishing industry struggles to effectively locate fish populations within the EEZ, as fishermen rely mainly on intuition and general migration patterns (Nordmo et al., 2022). While suitable for long-term planning, this approach falls short for daily operations, as fish can move significantly in a single day, requiring improved predictions. This challenge is becoming increasingly problematic as fish migration patterns become even more unpredictable due to climate change. Climate change affects both biotic and abiotic factors, such as sea surface temperature, which seems to have changed the timing of fish migrations (Kanamori, Yano, Okamura, and Yagi, 2024).

The lack of sufficient understanding of fish locations exacerbates the climate change problem, creating a reinforcing cycle that further harms the environment. Vessels often search for days or weeks before finding the target species and making a catch, consuming 3,000-5,000 liters of fuel per day and emitting about 5,000 kg of CO2 (Nordmo et al., 2022). Additionally, when target species are not found for longer periods, methods like bottom trawling, commonly used in Norway, can cause considerable bycatch and overfishing. This method involves dragging a net across the seabed, unintentionally capturing non-target species. Prolonged searches for target species increase bycatch, raising the risk of depleting key species and disrupting the ecological balance (Garrido and Starkey, 2020).

This study investigates whether data-driven techniques based on historical data can make an important contribution to reducing the environmental impact of the Norwegian fishing industry and helps to achieve more sustainable fishing methods. Sustainable fishing aims to reduce CO2 emissions by minimizing fuel consumption and ensures the long-term viability of fish populations by limiting overfishing and bycatch.

This research is part of the FishAI: Sustainable Commercial Fishing Challenge at the Nordic AI Meet, organized by NORA, addressing the urgent

need for innovative sustainable fishing practices in Norway (Nordmo et al., 2022). The FishAI competition aims to improve fishing sustainability in the Norwegian EEZ using advanced data-driven techniques. These data-driven techniques will be used to develop predictive models based on historical data, including catch note data and environmental data, both of which are spatio-temporal data sources. These models will predict high-likelihood fishing locations for commercially valuable fish species in Norway, aiming to identify the best fishing spots for each day of the upcoming fishing week to optimize catch efficiency.

1.2 State of art

Previous studies in the NORA FishAI competition investigated possible solutions to the Norwegian fishing sector's sustainability concerns. The research conducted by Lambon (2022), aimed to predict likelihood of fish presence of ten commercially valuable fish species within the Norwegian Exclusive Economic Zone (EEZ) at specific coordinates and provide user-friendly visualizations for fishermen. By utilizing environmental data, historical catch notes, and coordinate data, the researcher developed a Random Forest Regression Model. Despite the model's ability to identify areas close to actual fishing zones, its predictions varied in precision. It predicted larger areas likely to have fish but failed to consistently pinpoint the exact locations with the highest catches. As a result, the model's accuracy was low. The study suggests incorporating additional environmental variables and exploring alternative advanced machine learning models.

In a similar vein, the research by Brekke (2022) aimed to enhance fishing efficiency through a web application called 'Lodestar'. This platform provides an interactive map that displays locations predicted to have a high likelihood of catching 1 of 10 selected commercially valuable fish species, and integrates real-time vessel tracking. Utilizing historical catch data and environmental variables, an XGBoost Regressor was employed to predict these locations, and route optimization was achieved by addressing the Capacitated Vehicle Routing Problem (CVRP). The model retrieved a low accuracy of 5%, a significant decrease compared to the baseline model's 67% accuracy. The study suggested that the low accuracy might be due to the complexity of predicting multiple species together, including Epipelagic and Mesopelagic species that live at different depths, exhibit distinct behaviors and respond differently to environmental factors. These differences could have complicated the prediction process.

In the project of Linkiö, Lahtinen, and Kolmonen (2022) different approaches were explored to forecast fishing locations based on historical catch data. The outcomes where that k-means clustering and afterwards solving the traveling salesman problem predicts the most reliable fishing locations and creates the most efficient fishing plans. The findings of the research revealed that, while developing fishing plans for broader regions was feasible, pinpointing a single best fishing location was complex. Performance metrics for the prediction model, including precision, recall, and F1-score, showed suboptimal performance; their values were closer to 0 than to 1, indicating the difficulties in obtaining high accuracy for predicting specific fishing locations.

In the final proposed solution for the FishAI challenge, Syms (2022) discusses using satellite ocean data to predict the volume and location of fish, thereby aiming to enhance fishing precision and efficiency. The study outlines the use of a random forest regression model to forecast locations of 10 commercially valuable fish species. The model did not accurately predict the exact locations of the highest fish quantities for specific species, often deviating by more than 1000 km from the actual fish location. Syms (2022) suggests incorporating additional oceanographic variables in future research.

The study by Kanamori et al. (2024) investigated the long-term changes in timing and geographic location of North Pacific spiny dogfish migration from 1972 to 2019. The Barrier model revealed that the geographic location of migration remained stable over the study period, while the onset of migration began approximately a month earlier after the year 2000. Sequentially, a gradient boosting machine learning model was employed to analyze how SST, depth, and magnetic fields influenced migration occurrence rates, demonstrating high predictive accuracy. The study attributed spiny dogfish migration timing shifts primarily to spatial and spatio-temporal changes in SST driven by climate change. The study did not predict fish locations or quantities, focusing solely on predicting occurrence at fixed locations. Furthermore, there was a need for more detailed partitioning of environmental factors into spatial, temporal, and spatio-temporal components to better understand their independent and combined influences on migration patterns.

Another key contribution to the field is the research conducted by Koul, Sguotti, Årthun, Brune, Düsterhus, Bogstad, Ottersen, Baehr, and Schrum (2021), which used dynamical-statistical models to forecast decadal changes in Atlantic cod biomass, focusing on North Sea and Northeast Arctic cod stocks within Norway's Exclusive Economic Zone (EEZ). Dynamical-statistical models in this research combine sea surface temperature (SST) forecasts from dynamical models with linear regression

to predict cod biomass. Three approaches were tested: two simple linear regression models (using SST and fishing mortality separately) and a multiple regression model combining both variables. The multiple regression model provided the most accurate predictions for Total Stock Biomass (TSB). Retrospective forecasts compared predicted values with historical data, showing high performance in predictions. Furthermore, the forecasts indicated a potential decline in North Sea cod due to cooling sea surface temperatures in specific areas of the North Sea. Though this may seem contradictory to global warming, it highlights the regional variability of climate impacts and the crucial role of SST changes on cod biomass. These results emphasize the importance of integrating SST data into fisheries management and highlight the effectiveness of linear regression for predicting cod abundance in the Norwegian EEZ. However, while suitable for long-term forecasts, these models were less effective for capturing short-term fluctuations. The study also did not predict biomass at specific locations, making the predictions not applicable to particular areas in the Norwegian EEZ. Lastly, expanding the scope of variables was suggested to refine predictions.

1.3 Contributions

This research evaluates the predictive performance of an ensemble Lasso regression model in predicting the locations and corresponding catch yield of Atlantic cod within the Norwegian Exclusive Economic Zone (EEZ). This study uses linear regression, motivated by its success in previous research predicting cod biomass in the Norwegian EEZ, as discussed in Section 1.2. Specifically, this study uses Lasso regression, a variant of linear regression, which will be compared to the results of standard linear regression. Lasso is well-suited for this dataset, which contains many zero values, as it shrinks low-impact feature coefficients and focuses on the relevant data.

A meta-model approach combines multiple Lasso models to predict different features for each predicted day separately, capturing spatial and temporal context, and improving predictions by combining weaker models into a stronger ensemble model. Focusing exclusively on Atlantic cod, given its abundance of data, helps overcome the complexities of multi-species data and allows for a more tailored analysis of the interactions between environmental variables and cod distribution patterns. Based on previous research outlined in Section 1.2 and suggestions from the NORA Fish AI challenge, this study selects key variables: sea surface temperature, salinity, moon phase, and historical catch notes.

The corresponding research question is: What is the predictive performance of an ensemble Lasso regression model in predicting the locations and corresponding catch yield of Atlantic cod within the Norwegian Exclusive Economic Zone (EEZ), utilizing historical catch data and environmental data?

To address the research question, the first phase involved developing a data processing pipeline to align datasets, ensuring each data point captured both spatial and temporal context, with every data point representing a grid cell within the Norwegian EEZ on a specific day. To capture temporal patterns, 10-day windows were used, consisting of 5 input days and 5 output days. The input days included environmental data and historical catch data for each grid cell, while the output days provided the target values, specifying recorded catches and their corresponding catch weights for each grid cell. The ensemble Lasso regression model predicted cod locations and catch weights based on input days, using patterns learned during training. Meta-models were developed for each output day and target feature (longitude, latitude, product weight) to incorporate spatial and temporal context into the predictions. Visualizations displayed predicted cod locations and associated catch weights on a map of the Norwegian EEZ. Although the model produced low MSE and R^2 values during training, suggesting potential underfitting, the visualizations still captured the general trends in daily cod distributions, providing a fairly coherent representation of the actual distributions.

2 Theoretical framework

2.1 Cod migration and selected features

Over the course of a year, the migration of Atlantic cod within the Norwegian EEZ can be broadly divided into distinct stages among three main populations: Northeast Arctic cod (NEA cod), coastal cod, and North Sea cod. NEA cod migrate to northern coastal spawning grounds, particularly around the Lofoten Islands, Vesterålen, and the Barents Sea coast, from January to April. After spawning, they migrate to offshore and more northern areas to feed, and they are primarily found in the northern Barents Sea. As summer ends, NEA cod migrate back towards southern and western coastal areas and return to their spawning grounds from November to January. Coastal cod remain in nearshore areas year-round with localized migrations, spawning and feeding in various coastal and fjord areas along the Norwegian coast throughout the year. North Sea cod primarily stay in the southern part of the EEZ, spawning in offshore

locations within the North Sea from January to April, then migrating to deeper offshore feeding areas. This migration allows for efficient reproduction and access to nutrient-rich feeding grounds for development and energy accumulation (Olsen and Gjøsæter, 2010). The distinct seasonal migration patterns for each cod population (e.g., spawning from January to April) emphasize the importance of considering the month of the year in your model. Additionally, the repeating yearly migration patterns suggest that historical catch locations and catch yields of cod in the Norwegian EEZ are important indicators for predicting future cod locations and catch yields.

Salinity, the concentration of salt in seawater, is another key factor influencing cod migration. High salinity enhances nutrient availability and boosts primary productivity, which increases food resources for cod and affects their feeding areas (Myers, Drinkwater, Barrowman, and Baird, 1993). Since cod migrate toward nutrient-rich feeding grounds, incorporating salinity in the model provides deeper insights into their migration patterns.

The moon phase, particularly the presence of a full moon, also plays an important role in fish migration. The full moon increases ambient light levels at night, making fish and their prey more visible, prompting fish to adjust their depth and horizontal position in the water to avoid predators (Cohen and Forward Jr., 2009). Additionally, tidal changes from the moon's gravitational pull can alter feeding grounds and habitat accessibility, prompting fish to move to areas with more food or safer spawning conditions (Milardi, Lanzoni, Gavioli, Fano, and Castaldelli, 2018).

Climate change is causing greater year-to-year variation in cod migration patterns, as highlighted by the research of Kanamori et al. (2024), which underscores its increasing influence. Similarly, the study by Sundby and Nakken (2008) examines the impact of multidecadal climate oscillations and climate change on NEA cod (subspecies of Atlantic cod) spawning habitats along a 1500 km stretch of the Norwegian coast within the EEZ. The study found that during warmer periods, Atlantic cod spawning grounds shifted more northward as cod prefer colder waters, typically found in northern regions. In colder phases, spawning grounds moved southward as those areas became suitable. These temperature shifts are influenced by multidecadal climate variations. In addition, since the 1980s, human-induced climate change has caused a continuous rise in sea surface temperatures, leading to an ongoing northward shift in spawning regions, with new spawning activity observed in East Finnmark since 2003. This highlights the need for adaptive management strategies in fisheries management to effectively track and respond to changing fish locations. The discussed studies of Kanamori et al. (2024) and Sundby and Nakken (2008) highlight how cod habitats and migration patterns shift in response to temperature changes, reinforcing the importance of including sea surface temperature as a key feature in the study.

Therefore, the final selected features for the study are: historical catch locations, catch weights at those locations, the month of the year, salinity, moon phase, and sea surface temperature. Historical catch data and the month of the year are important for understanding yearly migration patterns, while the selected environmental factors influence these patterns, making it essential for the model to account for how changes in these conditions impact migration behavior.

2.2 Multi-output Linear regression

Linear regression is a statistical method used to understand the relationship between the output y and the features' inputs $x_1, x_2, x_3, \ldots, x_n$. This relationship is shown as

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p + \epsilon, \quad (2.1)$$

where β_0 is the intercept, $\beta_1, \beta_2, \ldots, \beta_p$ are the coefficients, and ϵ represents the error term. In linear regression, the dataset typically consists of multiple data points, each with different values for the variables. The goal of the model is to learn the coefficients (β) that best capture the underlying relationship between the independent variables and the dependent variable across all these data points. The model finds the coefficients that minimize the sum of squared residuals (RSS), shown in (2.2). Residuals, represented by the error term (ϵ) in (2.1), are the differences between the observed and predicted values. Minimizing the RSS means reducing the sum of the squares of these error terms, thereby minimizing the total error in the model's predictions. The RSS is a loss function and defined as

$$\text{RSS} = \sum_{i=1}^{m} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{m} \left(y_i - \left(\beta_0 + \sum_{j=1}^{n} \beta_j x_{ij} \right) \right)^2, \quad (2.2)$$

where y_i represents the actual value, and $\hat{y}_i$ is the predicted value, i is the index for the data points and j is the feature index. There are different methods to update the coefficients (β) to minimize the RSS, with gradient descent being a common approach, including in Lasso regression. The process starts with an initial guess for the coefficients, often small or random values, and in each iteration, they are adjusted in the direction that reduces the RSS. This adjustment is based on the gradient (slope) of the RSS with respect to each coefficient. The

coefficients are updated step by step until the RSS converges or the changes in the coefficients become small enough to stop.

Multi-output linear regression, also known as multivariate linear regression, expands upon simple linear regression by covering scenarios with multiple dependent variables. In multi-output linear regression, multiple dependent variables are predicted simultaneously. The relationship is represented in matrix form as:

$$\mathbf{Y} = \mathbf{XB} + \mathbf{E}, \tag{2.3}$$

where $\mathbf{Y}$ is a $n \times m$ matrix of observed values for m dependent variables across n observations. Each column in $\mathbf{Y}$ represents a different dependent variable. $\mathbf{X}$ is a $n \times (p+1)$ matrix of independent variables, $\mathbf{B}$ is a $(p+1) \times m$ matrix of coefficients, and $\mathbf{E}$ is an $n \times m$ matrix of errors. The matrix multiplication $\mathbf{XB}$ combines all independent variables with their respective coefficients, producing the predicted values $\hat{\mathbf{Y}} = \mathbf{XB}$. The observed values $\mathbf{Y}$ are the sum of the predicted values and the error term $\mathbf{E}$, shown in (2.3). During training, the coefficients in $\mathbf{B}$ are adjusted, as in single-output linear regression, to minimize $\mathbf{E}$. This approach allows us to handle multiple outputs simultaneously (Phillips, Heaney, Benmoufok, Li, Hua, Porter, Chung, and Pain, 2022).

This study employs multi-output linear regression to predict three dependent variables—longitude, latitude, and product weight—based on independent variables, among which are salinity, sea surface temperature (SST), and moon phase.

2.3 Multi-output Lasso regression Gradient regression

Multi-output Lasso regression maintains the same linear modeling framework as shown in (2.3) but adds an L1 regularization term for the coefficients to the loss function. This regularization encourages a sparse model by shrinking less important coefficients to zero. The objective function of multi-output Lasso regression is defined as:

$$\begin{aligned}\text{Loss} = &\sum_{k=1}^{K}\sum_{i=1}^{m}\left(y_{ik} - \left(\beta_{0k} + \sum_{j=1}^{n}\beta_{jk}x_{ij}\right)\right)^2 \\ &+ \lambda\sum_{k=1}^{K}\sum_{j=1}^{n}|\beta_{jk}|\end{aligned} \tag{2.4}$$

In this formula:

- i represents the index of observations ($i = 1, \ldots, m$).
- j represents the index of features ($j = 1, \ldots, n$).
- k represents the index of output variables ($k = 1, \ldots, K$).
- λ represents the regularization parameter controlling penalty strength on the coefficients.

The second term in this formula represents the L1 norm. A coefficient β_{jk} is set to zero when $|\beta_{jk}| < \frac{\lambda}{2}$, meaning its absolute value is smaller than half the regularization parameter. The value of λ determines the penalty strength, where a higher λ results in stronger regularization, increasing the likelihood of coefficients with low predictive power being set to zero.

Multi-output Lasso regression is particularly useful for datasets with many features and a significant number of zero values. In this study, some variables, particularly catch weight, contain a high proportion of zero values. All variables are linked to specific locations within the Norwegian EEZ, and many locations consistently report zero catch weight across all the days in the dataset. Lasso regression is well-suited to handle the sparse nature of such data because it applies L1 regularization, which shrinks the coefficients of less relevant features to zero. This process reduces model complexity and ensures that the model focuses on the most important features, ignoring those that contribute little to prediction accuracy, such as locations consistently reporting zero catch weight.

3 Methods

3.1 Data Description

Drawing from the conclusions in Section 2.1 regarding the important factors influencing cod migration, the following features were selected: historical catch locations, catch yields at those locations, the month of the year, salinity, moon phase phase, and sea surface temperature. Data for these features were retrieved from 4 datasets provided by the NORA FishAI competition: the catch notes dataset, sea surface temperature dataset, salinity dataset, and moon phase dataset (NORA, 2022). These datasets are good resources for addressing the research problem due to their comprehensive and clearly structured data. The datasets can be accessed through the following link: https://tinyurl.com/54w5bvxa/. The sea surface temperature (SST) dataset is made available by the National Oceanic and Atmospheric Administration (NOAA) (Oceanic and Administration, 2024). It provides global daily averages of the sea surface temperature from the year 1981 to present. This dataset is derived from satellite observations and is available at a resolution of 0.25

degrees latitude by 0.25 degrees longitude. The dataset contains a single variable, Sea Surface Temperature (SST), which is indexed by three dimensions: longitude, latitude, and time. The salinity values used in this study are sourced from the SMAP Salinity Version 4 dataset (2024) (SMAP). The dataset consists of global monthly salinity averages from 2015 to the present. It uses the same longitude x latitude resolution as the SST dataset, with a 0.25 by 0.25 degree grid. The dataset includes multiple variables indexed by three dimensions: time, longitude, and latitude. The primary variables are two types of salinity data: 40-km resolution (**sss_smap_40km**), which provides detailed salinity data at a finer scale, and the 70-km smoothed product (**sss_smap**), which is an averaged version representing the main salinity variable. Other variables provide information about related oceanographic conditions, such as the number of observations (**nobs**), the uncertainty estimate at 70-km resolution (**sss_smap_uncertainty**), and the gain weighted ice fraction per grid cell (**gice**). The moon phase data consists of dates and exact times of full moons from 1900 to 2050, specified in Central European Time (CET) (Chemkaeva, 2020). Historical catch notes, published by the Norwegian Directorate of Fisheries, are daily logged records from fishermen operating vessels larger than 15 meters within the Norwegian EEZ (Directorate, 2024). Spanning from 2000 to 2024, these catch notes provide information of each catch, with approximately 130 features per record and around one million notes per year. These features include, among others, the latitude and longitude coordinates of each catch, as well as information on quota type, fishing gear used, product weight, and species type. The catch notes also provide temporal context, as for each catch the 'Landing Date' is recorded. The Norwegian EEZ is defined by coordinates ranging from -3.125 to 36.625 degrees longitude and 56.125 to 74.625 degrees latitude. While the catch notes cover this entire EEZ, they also extend beyond its boundaries into adjacent maritime areas, including parts of the North-East Atlantic and Arctic Maritime Region. The notes specify coordinates for both the exact catch locations and the main fishing areas, with 12 main areas defined within this geographic extent.

3.2 Data preprocessing

3.2.1 Data Cleaning

The first step in the data cleaning process was selecting all relevant features from the dataset, which involved removing many unnecessary features. For the salinity data, only the 70-km smoothed salinity variable (**sss_smap**) was selected. Smoothed data reduces random variability and provides more consistent, less noisy values, making it the preferred salinity variable. For the catch notes, the only variables considered relevant were: product weight, landing date, longitude, latitude (precise coordinates compared to the main area coordinates), and species type. Data points for species other than cod were removed, allowing dropping the species column after filtering. Product weight, which is the weight after processing the fish, was selected over gross weight because it more accurately represents the fish catch. Gross weight includes non-fish weight such as seaweed. For the sea surface temperature, there were no variables removed as there was only one variable indicating the sea surface temperature. The moon phase data was filtered to include only the dates of full moons, with the time feature removed. A new binary feature was added to indicate the occurrence of a full moon, with a value of 1 representing a full moon.

In the second step, the datasets were synchronized to cover the same time frame by converting all sets into daily data points from the year 2015 to the end of 2021. This was the only period where all datasets had data points available. For datasets like salinity, which originally had monthly averages, the monthly average was copied to every day in that month to create daily data points. For the moon phase dataset, the dataframe was extended by filling in all days from 2015 to 2021 and assigning a value of 0 to days without a full moon. SST and catch notes already had daily data points. NaN values, representing non-numerical data, could not be processed by the Lasso regression model and therefore had to be replaced with numerical values. NaN values for salinity and product weight were filled with zeros. Most of the NaN values for salinity occurred on land, where salinity is naturally zero due to the absence of water. Therefore, it was logical to set the NaN values to zero. For product weight, the presence of NaN values introduced uncertainty about whether catches were reported. Since most product weight values in the EEZ were zero, replacing NaN values with zeros was considered appropriate. This prevents giving false importance to locations based on uncertain catch data. Missing months in salinity were replaced with values from the same month of the previous year, as the absence of data in those months indicated unreported values rather than the absence of salinity. This process ensured no datasets had missing values or days from the year 2015 to 2021.

Following that, the datasets needed to be spatially aligned. The dataset's coordinates varied in resolution: salinity and SST data used 0.25-degree increments, while catch notes had unstructured coordinates. To synchronize these datasets, a 0.25 x 0.25-degree grid was established using the resolution of the SST and salinity data as reference.

For latitude, a 0.25-degree increment is equivalent to roughly 27.75 kilometers, which is constant throughout the world. Depending on the exact latitude value, a 0.25-degree increment of longitude within the Norwegian EEZ ranges from roughly 11 to 18 kilometers. Initially, the values of SST and salinity were connected to coordinates indicating the lower-left corners of the grid cells. These coordinates were then transformed to the midpoints of the grid cells to provide a more central representation of each grid cell with respect to its coordinates. Catch notes were aggregated by summing product weights for data points within the same defined grid cell, resulting in a single data point per grid cell per day, ensuring alignment. The dataframes were then restricted to the Norwegian EEZ, which spans a longitude range from -3.125 to 36.625 degrees and a latitude range from 56.125 to 74.625 degrees, as they originally extended beyond this region (Flanders Marine Institute, 2023).

The different dataframes were aligned with respect to dates and coordinates to create the merged dataframe. This alignment ensured that the landing date, coordinates of the grid cell midpoint, salinity, SST, product weight, month (indexed from 1 to 12), and full moon indicator were recorded for every day from the beginning of 2015 to the end of 2021 for each grid cell in a tabular format.

Due to challenges in training the model with such a large-scale dataframe, the size of the dataframe needed to be reduced to improve training efficiency. The focus was placed on data points with positive product weights, as they are considered better predictors for the future presence of cod compared to data points with no catches. Consequently, not all grid cells of the Norwegian EEZ were covered for every day in the final dataframe. On the day with the highest number of positive grid cells, 112 out of 12,000 grid cells showed positive product weight values. Consequently, 112 grid cells were selected for each day. If fewer than 112 grid cells had positive product weights, additional grid cells with zero product weight values were randomly chosen for that day to ensure a consistent number of data points across all days. This reduced the dataframe from 12,000 to 112 data points per day. Afterwards, the data points in the dataframe were sorted in ascending order, first by time, followed by longitude, and then latitude, to maintain a coherent spatial and temporal structure. A reference to the first 4 rows of the final dataframe is provided in Table 3.1.

Table 3.1: First Rows of the Final Dataframe

Date	Lon	Lat	Weight	Moon	SST	Sal	Month
2015-01-01	4.50	57.75	1071.0	0.0	8.03	0.92	1
2015-01-01	6.50	57.75	4803.0	0.0	7.63	0.89	1
2015-01-01	17.50	69.75	4483.0	0.0	7.04	0.00	1
2015-01-01	18.00	74.75	0.0	0.0	4.90	0.88	1

3.2.2 Splitting and normalizing

Then the data had to be splitted in training, validation and testing sets. The training data will be used to train the model on the data to learn the patterns, relationships and features in the dataset. The validation dataset will be used to evaluate the model on unseen data, assess the chosen hyperparameters, and optimize them if necessary. The test data will be use to have an unbiased evaluation of the model after training and hyper-parameter tuning with unseen data. The dataset is split into the following ratios: 70% for training data, 20% for validation data, and 10% for testing data. To ensure consistency, the data was split at daily boundaries, maintaining 112 grid cells for each day in every dataset.

After splitting the data, the datasets were normalized by min-max normalization. This was done to prevent features dominating the learning process for the model due to differences in scale. Min-max normalization scales all features, except for the landing date, to a range between 0 and 1. The feature 'Landing Date' represents a specific point in time with multiple components (year-month-day), which is not linearly meaningful for normalization. Other features were scaled proportionally within the specified range, preserving the original relationships and distributions within the data, which is useful for maintaining the inherent characteristics of the dataset.

3.3 Labeling

Data labeling in this study is done using a sliding window technique to create input-output pairs from the temporal data. This approach divides the data into overlapping windows, allowing the model to learn temporal dependencies and patterns by linking past data points to future outcomes. This method is particularly useful in this study, which analyzes spatio-temporal data, as they help the model capture how environmental features and cod distributions change over time.

Each window consists of a set number of input and output time points. The window slides over the dataset with a predetermined offset, ensuring all time points are captured.

Consider a dataset D, where each data point is indexed by a time point t. A window W_i captures both past and future time points relative to a reference time point. If m input time points and n output time points are defined, with the reference time point set at t, the window W_i is defined as:

$$W_i = [x_{t-m}, x_{t-m+1}, \ldots, x_{t-1}, y_t, y_{t+1}, \ldots, y_{t+n-1}],$$

where $x_{t-m}, \ldots, x_{t-1}$ correspond to the input values (past data), and $y_t, \ldots, y_{t+n-1}$ correspond the output values (future data).

The next window W_{i+1}, when the offset is set to 1, has the new reference point at $t+1$, and is represented as:

$$W_{i+1} = [x_{t-m+1}, \ldots, x_t, y_{t+1}, y_{t+2}, \ldots, y_{t+n}] .$$

The window function is applied to the training, validation, and test datasets, organizing the data into input-output pairs. Consequently, the data is divided into $X_{\text{train}}, Y_{\text{train}}, X_{\text{val}}, Y_{\text{val}}, X_{\text{test}}, Y_{\text{test}}$, where X contains the input parts from different windows, and Y contains the corresponding output parts.

In this study, 10-day windows are used, covering 112 grid cells per day, totaling 1,120 data points per window (560 for input, 560 for output). This approach enables the model to predict the next 5 days based on the preceding 5 days after training. The choice of a 5-day input and output period is made because it effectively balances the need for sufficient days to capture temporal patterns while keeping the data dimensionality manageable.

The windows have a 1-day offset, meaning the reference time point shifts by 1 day (112 data points) for each subsequent window.

Each grid cell for every day in the window is associated with multiple features. Both the input and output parts of the window have the following 3-dimensional structure:

$$(\text{Windows}, \text{Days} \times \text{Data Points}, \text{Features})$$

Each window is indexed in the first dimension, with the second dimension including the 560 data points and the third dimension comprising the features. For the input datasets (X), all selected features except 'Landing Date' were included, as it cannot be normalized. The date of the reference time point of each window is stored separately in an array, which is indexed to map windows to their exact dates. The output datasets (Y) include only the features to be predicted: longitude, latitude, and product weight.

To start creating the input-output pairs with the window function, the first window begins 6 days after the initial day of the time range (January 1, 2015). This is because each window requires 5 input days and 5 output days, so the reference point starts after the first 5 days. The reference point consistently represents the first day of the output window. The window function then slides over the datasets with a 1-day offset. The window function stops this process when only 5 days remain to ensure complete input-output pairs.

3.4 Data Flattening Challenges

The input and output datasets must be flattened from a 3-dimensional format to a 2-dimensional format in order to train the Lasso regression model. As the regression model needs a two-dimensional input and output structure, flattening the datasets is crucial. This transformation results in the X and Y datasets being restructured as:

$$(\text{Windows}, \text{Days} \times \text{Data Points} \times \text{Features}).$$

The windowing and flattening process for both input and output data presented challenges in preserving the temporal and spatial structure in the data. The original 3 dimensional format of the windowed datasets, with dimensions: $(\text{Windows}, \text{Days} \times \text{Data Points}, \text{Features})$, already made it challenging to maintain temporal context because the different days were merged into a single dimension (the second dimension) without explicit labeling of distinction of days. Flattening this 3D dataset into a 2D dataset increased this problem, where each window instance in the output dataset is structured as:

$$n_i \in N : N = \text{All windows}$$
$$n_0 = \begin{bmatrix} \text{lat\_cell1\_day1}, \\ \text{lon\_cell1\_day1}, \\ \text{weight\_cell1\_day1}, \\ \ldots \end{bmatrix} . \quad (3.1)$$

In this flattened format, each row combines all feature values from different grid cells and days into a single dimension. The second dimension of the output datasets contains $5 \times 112 \times 3$ data points, representing 5 days with 112 grid cells and 3 feature values per cell. This restructuring means that each unique combination of a target feature (longitude, latitude, or product weight) with a specific grid cell and day is treated as an individual feature by the model during training, substantially increasing the number of features.

Furthermore, this format obscures the association of initial features (e.g., longitude, latitude) with specific grid cells and days, as these features are now integrated into composite features that combine grid cell, day, and feature type. This means that the integration results in a complete loss of spatial and temporal context in the data. As a result, the model predicts data points from the same days and locations as independent, without considering their relationships in time and space. This loss of spatial and temporal context in the data during both learning and predicting could lead to less accurate predictions that do not reflect real-world spatial-temporal patterns.

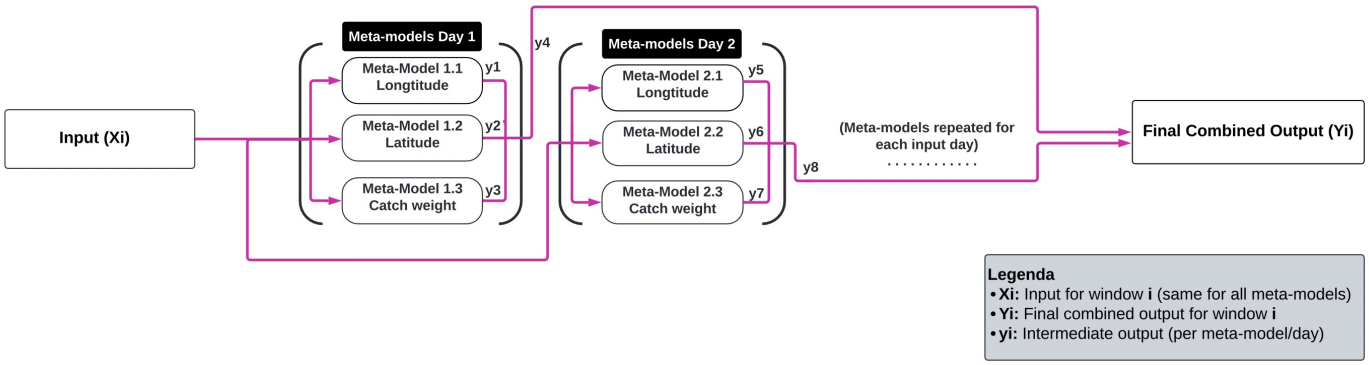

Figure 3.1: Composite Meta-Model Architecture

3.5 Modelling approach

3.5.1 Meta-models

To address the lack of spatial and temporal context in the data, the output data was divided into subsets, each containing a single target variable (product weight, longitude, or latitude) for one of five output days. This segmentation led to a model architecture comprising multiple meta-models, each trained on a specific target variable for a specific day. By separating target features, each data point in the output for a given window instance corresponds to a unique location, enhancing spatial context. This structure provides a location-specific organization for each data point, unlike the previous data format where spatial distinctions were obscured. Additionally, this setup allows the model to analyze relationships between location-specific variables (longitude and latitude) and input data independently of product weight, further reinforcing spatial context. Temporal context is provided by training each meta-model on data from a single output day, creating temporal distinctions. Thus, by training each meta-model separately on a single feature for a specific day, the spatial-temporal context is preserved throughout both learning and prediction processes.

Then the overall ensemble model, consisting of the individual meta-models, predicts the 5 output days for a specific window based on the 5 input days of that same window. A function is implemented that splits the target data into separate days for each window. Afterwards, the target data is converted to a 3D format. This conversion is important as it enables the model to be trained on distinct features (longitude, latitude, and product weight) for each specific day within a window. After conversion, the data can be separated by feature, as the third dimension subsequently indexes the different feature types. Each Lasso model is trained to predict its respective feature for the specified output day, using historical catch and environmental data from the input days, independently from the other features and days. The independent predictions for each feature are combined in the initial order: (lon_cell1_day1, lat_cell1_day1, weight_cell1_day1, lon_cell2_day1, ...), to retrieve daily predictions. The 5 different daily predictions are then combined in the correct order into a single long array for the specific window. All these predictions for the entire dataset, across all windows, will be combined into a single 2D dataset. This is done to effectively compare it with the target data, as the statistical measurements used in this study are only possible on 2-dimensional data.

An illustration of this model architecture is shown in Figure 3.1. This illustrates how the same input data flows through the different meta-models. Then, the different outputs are combined to a single array containing the predictions for the specific window indexed by i.

To compare the performance of Lasso regression with a baseline model, a standard linear regression composite model is implemented using the same meta-model framework. The theory behind standard linear regression is detailed in section 2.4.

Both the linear regression and Lasso regression models are implemented in Python using the Scikit-learn library (Pedregosa, Varoquaux, Gramfort, Michel, Thirion, Grisel, Blondel, Prettenhofer, Weiss, Dubourg, Vanderplas, Passos, Cournapeau, Brucher, Perrot, and Duchesnay, 2011).

3.6 Training and Metrics

The ensemble model is trained by using the input and output training datasets, X_train and Y_train. Different meta-models for every day and every target variable are initialized. The meta-models are implemented using the following Lasso regression parameters:

- α - Regularization strength, controlling the degree of L1 regularization.

- `max_iter` - The maximum number of iterations.
- `random_state` - A fixed seed value to ensure reproducibility.

The maximum number of iterations is set to 2000. This choice is computationally feasible and aligns with common practice, providing a good balance between convergence and computational efficiency. The random state is set to 42, a widely accepted and recognizable choice. The α parameter controls the degree of regularization of the Lasso regression. As this value influences generalization and model complexity, it can have a big impact on the predictions. That is why grid search will be applied to find for every meta-model the optimal alpha value. The parameter grid evaluated during the grid search is defined as follows:

- `param_grid = {'alpha': [0.001, 0.01, 0.1, 1.0]}`

The chosen values for the alpha parameter grid, span a range from low to high regularization strengths commonly used in Lasso regression. A function was created which iterates over all days and features, initializing a unique Lasso regression model for each. After initializing the model with the specified random state and maximum iterations, grid search with cross-validation is applied to the parameter grid. **GridSearchCV** is a scikit-learn class that is used to perform the grid search with cross-validation in order to find the optimal alpha values for the models (Pedregosa et al., 2011). It evaluates the effect of different alpha values using Negative Mean Squared Error (MSE) as the performance metric to identify the alpha that minimizes this error for each model. The use of negative MSE is necessary because the **GridSearchCV** class maximizes the scoring metric. Negating the MSE effectively transforms the objective to minimize it. MSE is particularly valuable for this study, which aims to predict cod locations (longitude and latitude) and their associated product weights. It captures the average squared difference between the predicted and actual values, providing a clear measure of the model's accuracy. By penalizing larger errors more heavily, it ensures the model focuses on making accurate location predictions.

An **GridSearchCV** instance is created for each combination of target day and target feature (meta-model), using the initialized Lasso model, parameter grid, 5-fold cross-validation strategy, and scoring metric as configurations. The 5-fold cross-validation helps prevent overfitting by exposing the model to different subsets of the training data. The grid search is applied by fitting the **GridSearchCV** instance. For each alpha value, the meta-model is trained on four folds and evaluated by the negative MSE of predictions on the fifth fold. This process is repeated for all folds, ensuring each fold serves as the validation set once. The average negative MSE across all folds is calculated, and the alpha that achieves the highest average negative MSE is selected as the optimal value for the meta-model.

After iterating over all the meta-models, the function returns the **best_alphas** array containing the best alpha values for each model. After the optimal alpha values are found, the final models are initialized with these alpha values and trained on the training dataset. The overall performance of the meta-models is evaluated based on predictions on the input datasets within the training, validation, and test datasets. The results of the training data predictions can indicate possible overfitting or underfitting. Validation data is used to fine-tune the hyper-parameter alpha if the model responds unexpectedly to unseen data. Testing data is used for a final, unbiased evaluation of the composite model.

Together with the MSE score, the R^2 of the predictions made on the training, validation, and testing datasets will be reported. The R^2 score indicates the proportion of variance in the dependent variables (e.g., product weight, longitude, latitude) predictable from the independent variables (e.g., SST, salinity, moon phase). R^2 offers information on the explanatory power of the model, whereas MSE assesses prediction accuracy.

3.7 Predictions visualization

After performance metrics are applied to the predicted outputs of the composite Lasso regression meta-model, visualizing the predictions helps to provide a clear and comprehensive understanding of location predictions. A function will visualize non-zero actual and predicted product weights for the 5 output days of the windows, using maps of the Norwegian EEZ. In these maps, each dot represents a predicted cod population location, with color variations indicating the magnitude of the product weight. By highlighting the differences between higher and lower product weights based on coordinates, this visualization helps fishermen better understand the predicted distribution of cod on specific days within the Norwegian EEZ. This visual comparison between predicted and actual cod locations, along with their associated product weights, aims to provide a more intuitive understanding of model performance beyond numerical metrics.

To facilitate visualization, it is necessary to denormalize the longitude, latitude, and product weight values for both the predicted and actual test datasets to display their real-world values. The de-

normalization process uses the following formula:

$$\text{data}[:,:,i] = \text{data}[:,:,i] \times (\max_i - \min_i) + \min_i, \tag{3.2}$$

Where i represents the feature index, $\max_i$ is the highest value of the feature, and $\min_i$ is the lowest value of the feature.

The test data must be converted for the denormalization into a 3D format to extract the separate features along the third dimension:

$$(\text{Windows}, \text{Days} \times \text{Datapoints\_per\_day}, \text{Features})$$

The function **visualize_specific_window** visualizes actual and predicted cod locations with their associated product weights for all the 5 output days within a specified window of the test dataset. First, it loads geographic data and filters it to include only coordinates within the longitude and latitude range of the Norwegian EEZ. For each output day within the window, the function calculates the start and end indices in both the actual and predicted output arrays to extract the relevant data for that day. Using window indexing, the same days of the predicted values and actual values can be visualized together. The exact date for each day in the output is calculated by adding the day index to the start date of the output days, with start dates for each window's output part stored in a separate array. Data frames are created for both actual and predicted data of the specific window, with columns for the features longitude, latitude, and product weight, to facilitate easier manipulation and plotting. Using longitude and latitude, these dataframes are transformed into geospatial data frames with the GeoPandas library, adding geometry information (points) to enable spatial operations (Jordahl, den Bossche, Wasserman, McBride, and Contributors, 2020). For example, a spatial join was applied to the geospatial data frames to identify and remove land points, ensuring that only points in the water appear in the visualizations.

In the final visualizations, a geographic map of Norway serves as the base layer, with scatter points representing cod locations. The color of each point reflects the product weight, with a colorbar indicating the scale: yellow dots represent higher product weights, while purple dots represent lower weights.

4 Results

The results focus on evaluating the predictive ability of the Lasso Regression Ensemble Model compared to a baseline model. First, grid search results will be assessed to identify which alpha values for the different Lasso models minimize the MSE of their predictions the most. The model's performance is assessed using the $\mathbf{R^2}$ score and MSE, derived from the predictions made on the training, validation, and test datasets. Furthermore, the visualizations of specific time windows will be analyzed to compare the distributions of cod predictions within the Norwegian Exclusive Economic Zone (EEZ) with the actual distribution patterns observed during those time intervals.

The grid search evaluated different alpha values for the different Lasso models to minimize the MSE of their predictions. The resulting best alpha values are stored in a 2-dimensional array, where the y-axis represents the days and the x-axis represents the three different target features. Therefore, the first value of the array (upper left corner) corresponds to day 1 - feature 1. The resulting array appears as follows:

$$\text{best\_alphas} = \begin{bmatrix} 0.001 & 0.1 & 0.001 \\ 0.001 & 0.1 & 0.001 \\ 0.001 & 0.01 & 0.001 \\ 0.001 & 0.01 & 0.001 \\ 0.001 & 0.01 & 0.001 \end{bmatrix}$$

All best alpha values for predicting longitude and product weight across the various days are set at 0.001, indicating a low level of regularization strength. This suggests that a lot of the coefficients provide sufficient explanatory power in predicting both longitude and product weight. The best alpha values for predicting latitude are higher, around 0.1 and 0.01. This implies that latitude relies on fewer relevant features, requiring more regularization to achieve optimal predictions. Higher alpha values for latitude, compared to the other target features, suggest that more features have limited predictive power for latitude, leading to their coefficients shrinking to zero. Although latitude and longitude are closely related, the features seem to describe longitude more effectively. Multiple factors may explain this, such as longitude showing more variability or being more strongly correlated with other features than latitude.

Table 4.1 compares the evaluation metrics (MSE and $\mathbf{R^2}$ scores) for both the baseline model and two different composite Lasso regression models across training, validation, and test datasets.

The R^2 score of the baseline model indicates a perfect fit on the training data ($R^2 = 1.000$), while the extremely low MSE suggests a high risk of overfitting. On the test data, the R^2 score drops to -4.026×10^{28}, indicating severe overfitting and poor performance on unseen data. The MSE on the validation and test data is 0.180, further indicating a poor model fit. Given the data is scaled between 0 and 1, this MSE is relatively high. Looking at the metrics of the composite Lasso regression model, there are remarkably different results compared to the baseline model. The composite Lasso regression model maintained a consistent MSE of 0.021

Table 4.1: Performance Metrics for Two Different Composite Lasso Regression Models and a Linear Regression Baseline Model

Model	MSE Training	MSE Validation	MSE Test	R^2 Training	R^2 Validation	R^2 Test
Lasso Regression Composite Model	0.021	0.021	0.021	0.046	0.019	-7.902×10^{25}
Lasso Regression Composite Model (α-values reduced by 10x)	0.016	0.023	0.024	0.340	-0.153	-7.902×10^{25}
Baseline Model	1.831×10^{-29}	0.180	0.180	1.000	-28.940	-4.026×10^{28}

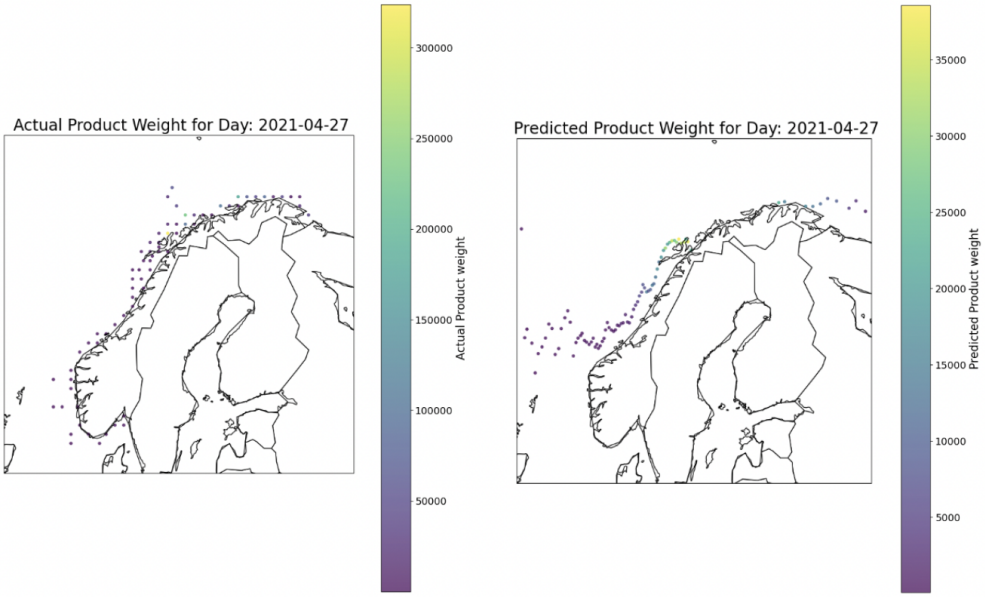

Figure 4.1: Predicted and Actual Product Weight for Day: 2021-04-27 (initial composite Lasso model)

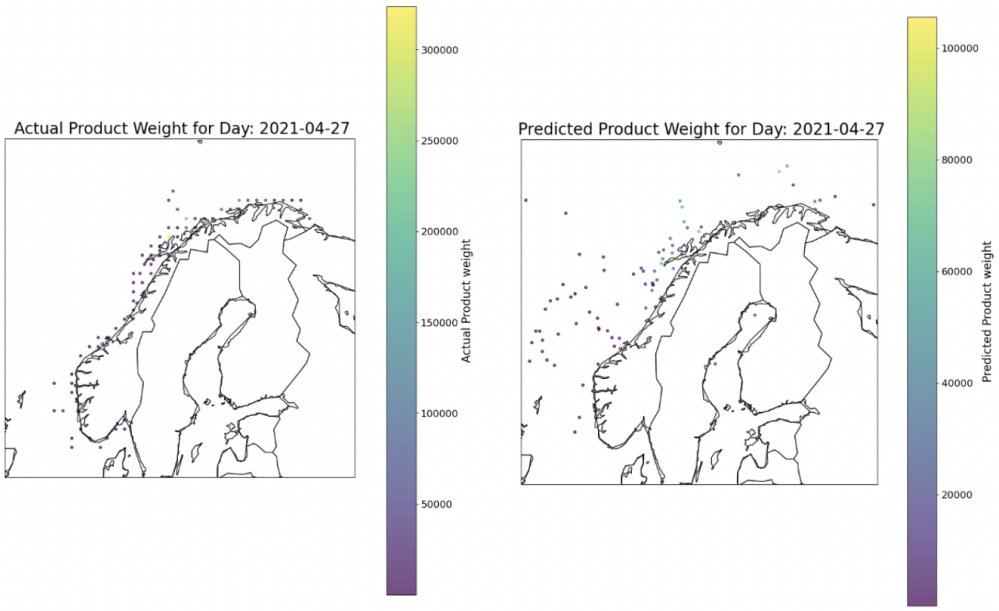

Figure 4.2: Predicted and Actual Product Weight for Day: 2021-04-27 (composite Lasso model, alphas reduced by 0.1)

across training, validation, and test datasets, indicating superior prediction accuracy and markedly better generalization without overfitting. Since the MSE on the validation set is nearly identical to that of the training set, further parameter tuning is not considered necessary. However, the R^2 on the training data shows that only 4.6% of the variance in the data is explained. On the test data, the R^2 is highly negative at -4.026 $\times$ 10^{28}. A negative R^2 score indicates the model performs worse

than simply predicting the mean of the test data. The model fails to capture the data's variance and produces larger errors than a simple benchmark. Considering these results, it is unusual and unexpected to observe a low MSE alongside a low $\mathbf{R^2}$ score. A possible explanation of these results is that the composite Lasso regression model is underfitting the data. The model makes predictions that are generally close to the mean, resulting in a low MSE. However, it fails to capture the data's variance and complexity, leading to a very low or negative R^2 score. This suggests the model's simplified predictions lack the variability needed for accurate, reliable estimations.

The potential underfitting issue becomes more apparent when comparing the visualizations of the predicted cod locations' distributions with the actual distributions. Appendix A.1 displays four days of predicted versus actual distributions for the first test window, while Figure 4.1 highlights one specific output day (2021-04-27) for this comparison. Across multiple windows, the predicted distributions consistently form an simplified S-shaped pattern, missing the finer details of the actual distribution. The oversimplified predictions may result from the Lasso model over-penalizing certain coefficients and shrinking them to zero, even though most alpha values apply relatively low regularization. This over-penalization can cause the model to ignore important features or data variations, leading to the simplified prediction patterns seen across all windows.

Despite this oversimplification and lack of precision in the predictions, the predicted versus actual fish distribution maps show that the model provides a fairly accurate overview of cod distributions across windows. The model captures key patterns, such as higher volumes in the north, indicated by more yellow dots.

To address the oversimplification issue, the strength of the alpha values was reduced by a factor of 10, resulting in the following alpha grid:

$$\text{best\_alphas} = \begin{bmatrix} 0.0001 & 0.01 & 0.0001 \\ 0.0001 & 0.01 & 0.0001 \\ 0.0001 & 0.001 & 0.0001 \\ 0.0001 & 0.001 & 0.0001 \\ 0.0001 & 0.001 & 0.0001 \end{bmatrix}$$

When the regularization was reduced by lowering the alpha values, the model captured more detail from the training data, as seen by the improved MSE in Table 4.1. However, this adjustment caused predictions to disperse more across the map rather than concentrate along the coast, which reduced alignment with the actual fish distributions, as shown in Figure 4.2 for the fourth output day of the first window (2021-04-27). While the model achieved a better fit to the training data, it struggled to generalize to unseen data, leading to a slight increase in MSE for the validation and test sets, with the validation $\mathbf{R^2}$ decreasing to a negative value (from positive in the initial composite Lasso model). Therefore, the initial alpha values seem to produce the most accurate representation of the actual fish distributions across all three tested models.

Another important point regarding the MSE score is its interpretation after denormalization. For the primary Lasso model tested, the denormalized MSE of 0.021 corresponds to deviations of approximately 92 km in latitude, 46 km in longitude, and 27,168 kg in product weight, given that the predictions were scaled from 0 to 1. Although the MSE of the model is relatively low on the normalized scale, the deviations increase substantially after denormalization, making the predictions potentially inadequate for real-world fishing practices.

In conclusion, while the Lasso Regression Ensemble Model seems to oversimplify, resulting in consistent S-shape distribution patterns, it still performs well in providing a fairly realistic overview of the actual fish distributions. The model captures key trends, such as higher fish volumes in the north and clustering of Cod locations along the coast. However, the model's precision appears limited, especially after denormalization of the MSE, as deviations increase substantially across the three target features. This suggests it provides a fairly realistic overview of daily cod distributions but may not yet capture the finer details needed for more accurate and reliable estimations of cod locations and associated product weights.

5 Discussion

The goal of this research was to evaluate the performance of a composite Lasso regression model in predicting high-likelihood locations of Atlantic Cod based on geospatial data. The model successfully captured general trends in daily cod distributions within the Norwegian EEZ, predicting higher volumes in northern regions and along the coastline. However, the predicted distributions consistently exhibited an S-shaped pattern with a leftward tail, likely caused by oversimplification from the regularization parameter, which reduced variability in the predictions. While reducing regularization eliminates the simplified S-shape, it leads to other issues, such as highly dispersed distribution patterns that do not align with coastal patterns. This suggests that oversimplification due to the regularization parameters isn't the only issue; other factors, such as data processing challenges or model selection, likely also contribute to the oversimplified distribution patterns and the lack of precision in its predictions.

5.1 Limitations

After analyzing the experimental setup, a major flaw likely contributing to the reduced prediction accuracy is the loss of spatial-temporal context due to data flattening, which was needed for training the composite Lasso model. As detailed in Section 3.4, flattening the input and output data formats into 2-dimensional formats for training caused a loss of distinction between grid cells and days, which disrupts the model's ability to capture essential spatial-temporal relationships. To address this, the output was restructured into subsets, with each meta-model trained on one day and feature to create spatial-temporal context in the output. However, for the model to learn the spatial-temporal relationships effectively between input and output, the input needs a similar structure. Without this alignment, the model lacks necessary context, hindering its ability to capture spatial-temporal patterns effectively.

Another limitation arises from the grid cell selection process, where each day includes only a subset of grid cells (112 grid cells). This approach, intended to reduce data volume, means the model only receives data from specific areas of the map each day. This incomplete spatial coverage prevents the model from learning broader spatial patterns and understanding inter-regional interactions over time.

Furthermore, the model fails to recognize the time-evolving nature of variables (e.g., sea surface temperature, salinity) connected to grid cells within the windows. Each variable associated with a specific grid cell creates five model features, one for each of the five input days. However, the model treats them as independent observations rather than recognizing the temporal relationships between them. For example, salinity values for a specific grid cell over five days should form a sequence, but the model fails to capture how each day's value influences the next. The model also overlooks the sequential relationship between input features and their corresponding output with respect to time. The model's inability to identify time-based sequential patterns in the input data, as well as the sequential relationships between the output and input data, limits its ability to effectively learn and predict time-series patterns.

Another limitation of this research is the choice of Lasso regression, which may pose issues due to its simplicity. Lasso primarily relies on linear relationships (Phillips et al., 2022), while geospatial analysis often involves non-linear relationships due to the complexity of spatial processes and interactions (Fotheringham, Brunsdon, and Charlton, 2002).

The study is further complicated by the temporal bias in the catch notes dataset, which spans 7 years. The research by Brekke (2022), one of the proposed solutions to the FishAI challenge, found that fishermen tend to return to recently visited locations but gradually shift to new spots over time. However, the model treats recent and older data equally, causing the model to overemphasize outdated patterns and struggle to capture ongoing trends. Additionally, disruptions like COVID-19 potentially introduced additional bias into the dataset, as the model was trained on data from that period, which may not reflect typical fishing patterns. To improve accuracy, models should prioritize recent data to better capture the gradual changes in fishing behavior and the productivity of locations. Lastly, the limited scope of independent variables presents a potential flaw in the research. The model, trained on approximately 200,000 data points, utilizes only 7 independent variables, which may be insufficient for capturing the complex migration patterns and increases the risk of underfitting.

5.2 Future Work

To address the limitations of the current model architecture, a few main improvements should be prioritized in a future model. First, to address the current model's limitations in capturing spatial-temporal context in learning and prediction, it is crucial to explore models that can incorporate this context effectively. Additionally, exploring models that can capture the complex, non-linear patterns commonly found in geospatial data is important, as Lasso regression's linear design cannot achieve this. Furthermore, a key improvement would be to find a model that can handle higher-dimensional data than the current model and process all grid cells in the daily maps, thereby eliminating coverage gaps caused by subset selection.

The Perceiver model, a neural network model, is a promising option for addressing these challenges. The model is designed to handle complex, high-dimensional data by using attention mechanisms and latent arrays to process only the most relevant information (Jaegle, Gimeno, Brock, Vinyals, Zisserman, and Carreira, 2021). This approach maintains scalability, reduces computational load, and is likely to allow for the inclusion of all grid cells.

Additionally, an important benefit of the Perceiver model is its ability to learn non-linear relationships from the data due to its neural network architecture, making it well-suited to the complexities of geospatial data (Jaegle et al., 2021).

One of the most important advantages of the Perceiver model is its use of spatial and temporal encoding to process complex, spatial-temporal data (Jaegle et al., 2021). Spatial encoding provides positional information to feature values, helping the

model understand where each value is located and enabling it to capture spatial relationships across different locations. Temporal encoding provides feature values with a structured representation of time, enabling the model to identify sequential patterns and time-based distinctions. Together, these encodings provide the model with the spatial-temporal context needed for successful learning and prediction.

Another area of future research involves identifying additional variables that could be incorporated to enhance the model's capacity to capture variability within the data. Investigating factors that explicitly influence the migration process of the Atlantic cod species would be valuable. By incorporating these variables, the research can be more tailored to address the unique ecological needs and behaviors of Atlantic cod. For instance, including the availability of key prey species such as Capelin (Mallotus villosus), which is crucial to Atlantic cod diets, could enhance prediction accuracy by providing a more specific analysis (Deng and Lumley, 2023).

Lastly, establishing a benchmark for evaluating the model's performance, potentially by gathering human-predicted fishing locations or heuristic approaches, would enable a more comprehensive comparison. This benchmark could provide valuable insights into the model's practical utility by revealing whether it outperforms human estimations of fish locations and how well it aligns with real-world decision-making processes.

6 Conclusion

This study focused on the performance of a composite Lasso regression model in predicting the locations and catch yield of Atlantic Cod in the Norwegian Exclusive Economic Zone. The model predicted fairly realistic cod distribution patterns, but underfitting was suspected as the predictions consistently followed a simplified S-shape across most windows. Although the MSE scores showed relative low prediction error on a normalized scale, low R^2 scores indicated poor variability capture, further supporting the presence of underfitting. Furthermore, when the MSE was denormalized to the original scales of the three target features, the resulting deviations were substantially larger than they appeared on the normalized scale. Based on these findings, the main conclusion is that while the visualizations show the model's ability to capture general trends in daily cod distributions within the Norwegian EEZ, further research is needed to assess whether the model's predictions—particularly considering the denormalized location and weight deviations—are useful for practical applications. The absence of temporal and spatial context in the input data is a critical limitation, as it prevents the model from effectively learning the spatiotemporal patterns in the data. Moreover, the choice of Lasso regression is not suitable for this study, as it is not able to capture the complex, non-linear relationships often inherent in geospatial data. Future work could explore advanced non-linear models that incorporate temporal and spatial encoding in order to better capture spatiotemporal patterns from the data. Furthermore, introducing a benchmark for comparison, such as evaluating the model's performance against human heuristics, would provide valuable insights into its applicability for practical fishing operations.

References

J. Hole K. Løddesøl L. Ortheden J. Roaldsnes. T Brekke, A. Dammen. The lodestar fishing platform. *Nordic Machine Intelligence*, 2: 10–12, 2022.

D. Chemkaeva. Full moon calendar 1900-2050. Kaggle, 2020. Retrieved September 20, 2024, from `https://www.kaggle.com/datasets/lsind18/full-moon-calendar-1900-2050`.

Jonathan H. Cohen and Richard B. Forward Jr. Zooplankton diel vertical migration—a review of proximate control. *Oceanography and Marine Biology: An Annual Review*, 47:77–110, 2009.

Norwegian Seafood Council. Norway exports second highest value of seafood ever in 2020, 2021. URL `https://www.seafood.no/en/news/2020-export-statistics`. Accessed: 2022-05-27.

Y. Deng and T. Lumley. Multiple imputation through xgboost. *Journal of Computational and Graphical Statistics*, pages 1–19, 2023. doi:10.1080/10618600.2023.2157084.

Norwegian Fishing Directorate. Catch notes dataset. `https://www.fiskeridir.no/Tall-og-analyse/AApne-data/Fangstdata-seddel-koblet-med-fartoeydata`, 2024. Accessed: 2024-07-18.

Flanders Marine Institute. Marine regions, 2023. URL `https://www.marineregions.org/gazetteer.php?p=details&id=5686`. Accessed: 2024-07-20.

A. Stewart Fotheringham, Chris Brunsdon, and Martin Charlton. *Geographically Weighted Regression: The Analysis of Spatially Varying Relationships*. Wiley, Chichester, UK, 2002.

Á. Garrido and D. J. Starkey. *Too Valuable to be Lost: Overfishing in the North Atlantic*

since 1880. De Gruyter Oldenbourg, Berlin, Boston, 2020. doi:10.1515/9783110641738. URL https://doi-org.proxy-ub.rug.nl/10.1515/9783110641738.

A. Jaegle, F. Gimeno, A. Brock, O. Vinyals, A. Zisserman, and J. Carreira. Perceiver: General perception with iterative attention. *Proceedings of the 38th International Conference on Machine Learning*, 139:4651–4664, 2021. URL https://proceedings.mlr.press/v139/jaegle21a.html.

Kelsey Jordahl, Joris Van den Bossche, Jeremy Wasserman, Jacob McBride, and GeoPandas Contributors. Geopandas: Python tools for geographic data, 2020. URL https://geopandas.org. Version 0.8.1.

Y. Kanamori, T. Yano, H. Okamura, and Y. Yagi. Spatio-temporal model and machine learning method reveal patterns and processes of migration under climate change. *Journal of Biogeography*, 51:522–532, 2024. doi:10.1111/jbi.14595.

V. Koul, C. Sguotti, M. Årthun, S. Brune, A. Düsterhus, B. Bogstad, G. Ottersen, J. Baehr, and C. Schrum. Skilful prediction of cod stocks in the north and barents sea a decade in advance. *Communications Earth & Environment*, 2, 2021. doi:10.1038/s43247-021-00206-3.

E. Saet M. Maranon Z. Berlin S. Lambon, A. Sagun. Fishmaze: Fish monitoring andai-based zone evaluation. *Nordic Machine Intelligence*, 2: 7–9, 2022.

V. Linkiö, K. Lahtinen, and J. Kolmonen. Clusters and traveling fisherman. *Nordic Machine Intelligence*, 02:13–15, 2022. doi:10.5617/nmi.9930.

M. Milardi, M. Lanzoni, A. Gavioli, E. A. Fano, and G. Castaldelli. Tides and moon drive fish movements in a brackish lagoon. *Estuarine, Coastal and Shelf Science*, 215:207–214, 2018. doi:10.1016/j.ecss.2018.10.021.

R. A. Myers, K. F. Drinkwater, N. J. Barrowman, and J. W. Baird. Salinity and recruitment of atlantic cod (gadus morhua) in the newfoundland region. *Canadian Journal of Fisheries and Aquatic Sciences*, 50(8):1599–1609, 1993. doi:10.1139/f93-181.

NORA. Sustainable fishing fatasets. https://tinyurl.com/54w5bvxa, 2022. Accessed: 2024-07-18.

T.-A. S. Nordmo, O. Kvalsvik, S. O. Kvalsund, B. Hansen, P. Halvorsen, S. A. Hicks, D. Johansen, H. D. Johansen, and M. A. Riegler. fishai: Sustainable commercial fishing challenge. *Nordic Machine Intelligence*, 2:1–3, 2022. doi:https://doi.org/10.5617/nmi.9657.

National Oceanic and Atmospheric Administration. Noaa oisst v2 high resolution dataset. https://www.psl.noaa.gov/data/gridded/data.noaa.oisst.v2.highres.html, 2024. Accessed: 2024-07-18.

Aanes S. Mehl S. Holst J. C. Aglen A. Olsen, E. and H. Gjøsæter. Cod, haddock, saithe, herring, and capelin in the barents sea and adjacent waters: a review of the biological value of the area. *ICES Journal of Marine Science*, 67:87–101, 2010. doi:10.1093/icesjms/fsp229.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

T. R. F. Phillips, C. Heaney, E. Benmoufok, Q. Li, L. Hua, A. Porter, K. F. Chung, and C. C. Pain. Multi-output regression with generative adversarial networks (morgans). *Applied Sciences*, 12(18):9209–9209, 2022. doi:https://doi.org/10.3390/app12189209.

NASA Soil Moisture Active Passive (SMAP). Smap salinity v4 dataset. https://salinity.oceansciences.org/data-smap-v4.htm, 2024. Accessed: 2024-07-18.

S. Sundby and O. Nakken. Spatial shifts in spawning habitats of arcto-norwegian cod related to multidecadal climate oscillations and climate change. *ICES Journal of Marine Science*, 65 (6):953–962, 2008. doi:10.1093/icesjms/fsn085.

C. Syms. Satellite ocean data can informprecision fishing. *Nordic Machine Intelligence*, 2:4–6, 2022.

A Appendix A

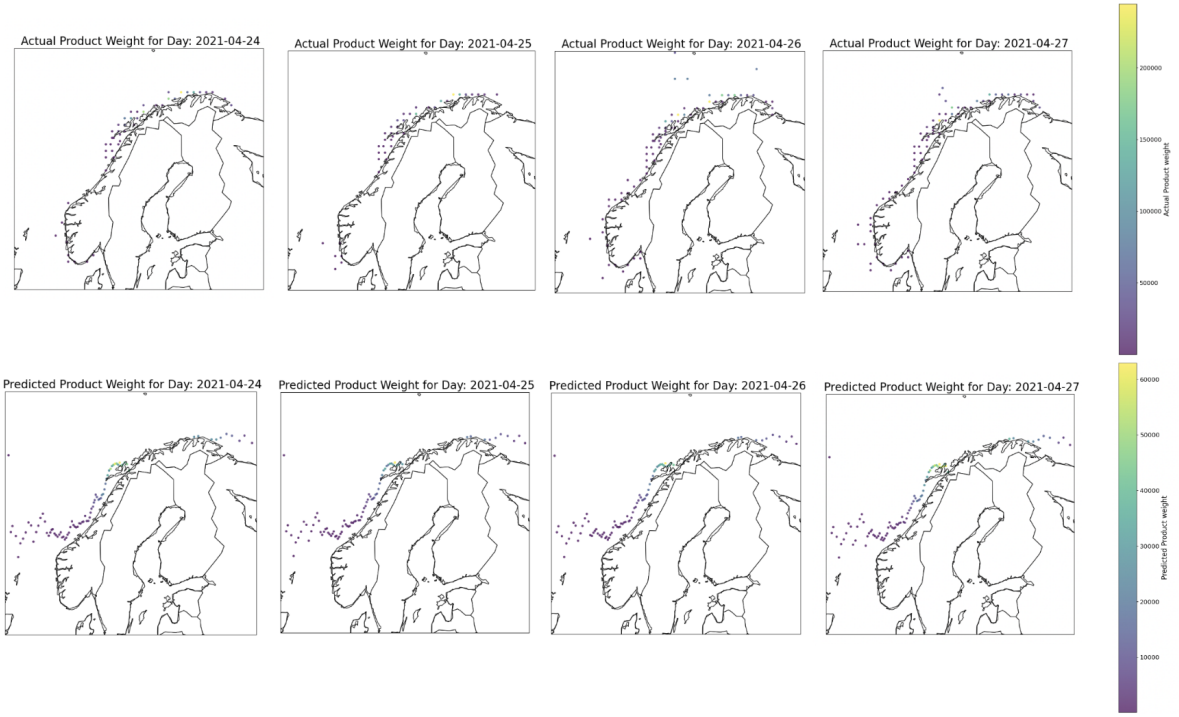

Figure A.1: Comparison of Actual vs. Predicted Fish Distribution for the 5 Output Days of the First Window (Window Index = 0)