Fall 12-18-2020

Machine Learning Model Selection for Predicting Global Bathymetry

Nicholas P. Moran
npmoran@uno.edu

Machine Learning Model Selection for Predicting Global Bathymetry

A Thesis

Submitted to the Graduate Faculty of the
University of New Orleans
in partial fulfillment of the
requirements for the degree of

Master of Science
in
Computer Science
Artificial Intelligence

by

Nicholas Moran

B.S. Southeastern Louisiana University, 2017

December, 2020

Acknowledgements

I would like to thank the UNO faculty of the Computer Science department for their stewardship over the past 3 years. I have learned much from many of the professors and enjoyed my time as a graduate student. Specifically, I want to thank Dr. Hoque for operating as my thesis advisor and providing excellent guidance. He kept a considerate attitude throughout the process and was always eager to help.

I would like to thank Dr. Warren Wood of the Naval Research Laboratory, Ocean Sciences Division. He has laid the groundwork for this thesis work as well as providing excellent guidance.

I would like to thank my co-scientists of the Naval Research Laboratory, Ocean Sciences Division. Specifically, Ben Stringer, Brittney Jarreau, Jenna Dauzat, Adam Minor, Elias Ioup, Norm Schoenhardt, and Bruce Lin. They have given much of their personal time to give constructive critiques of my paper and methods. As well as, offer excellent direction and suggestions for improving my results. This thesis would not have been possible without them.

Finally, I want to thank my grandmother Louise Moran. She has been my counsel, cheerleader, and console. Her prayers and gentle voice are a comfort I hold very close to my heart.

Contents

List of Tables

List of Figures

Acronyms

MBES Multi-Beam Echo Sounder

SBES Single Beam Echo Sounder

EGM Earth Gravitational Model

SDB Satellite-Derived Bathymetry

NAVO Naval Oceanographic Office

GEBCO General Bathymetric Chart of the Oceans

JAMSTEC Japan Agency for MarineEarth Science and Technology

NOAA National Oceanic and Atmospheric Administration

NGA National Geospatial Agency

RMSE Root Mean Square Error

MLP Multi-Layer Perceptron

ML Machine Learning

ETOPO Earth TOPOgraphy

NGDC National Geophysical Data Center

API Application Programming Interface

ANN Artificial Neural Network

Abstract

This work is concerned with the viability of Machine Learning (ML) in training models for predicting global bathymetry, whether there is a best fit model for predicting bathymetry, and optimizing bathymetry predictions. The desired result is an investigation of the ability for ML to be used in future prediction models and to experiment with multiple trained models to determine an optimum selection. Ocean features were aggregated from a set of external studies and placed into two minute spatial grids representing the earth's oceans. A set of regression models, classification models, and a novel classification model were then fit to this data and analyzed. The novel classification model is optimized by selecting the best performing model in a geospatial area. This optimization increases prediction accuracy for test purposes by approximately 3%. These models were trained using bathymetry data from the ETOPO2v2 dataset. Analysis and validation for each model also used bathymetry from the ETOPO dataset, and subsequent metrics were produced and reported. Results demonstrate that ocean features can potentially be used to build a prediction model for bathymetry with the inclusion of accurate data and intelligent model selection. Based on the results in this work, evidence supports that no single model will best predict all Global bathymetry and choosing optimum models for areas will benefit prediction accuracy.

Machine Learning, Bathymetry Prediction, Geo-Physics, Model Selection

1 Introduction

Bathymetry is defined as the measure of depth in a body of water. The forefront in global bathymetry mapping is the aggregation of predicted and measured sources. The predictions come from models called EGM. These Earth Gravitational Model (EGM) are the standard for measuring global bathymetry [17, 6, 9, 20]. They predict bathymetry by modeling the relationship between the vertical gravity gradient of a geoid and the depth of a water column. This relationship is modeled by measuring the altimetry of the sea surface to a constellation of satellites. Altimetry is the measurement of height or altitude. Specifically, this paper is referring to the altimetry of the sea surface using the distance to a satellite. Measuring the altimetry can lead to the identification of swells that are caused by gravity gradients that originate from geoids. Modeling the resulting bathymetry as a function of the approximated gravity is an excellent approach at coarse resolutions, but at finer resolutions, the prediction error becomes an issue. Modern EGM predictions have an error of approximately 180 meters. On the other hand, sonar platforms such as the Multi-Beam Echo Sounder (MBES) [1] provide extremely accurate measurements of depth. They operate using the simple physics of sound traveling through the medium of water. Specifically, they operate by measuring the time a sonar ping requires to travel to the bottom and back to a hull-mounted sensor. This measurement of time is used along with sound speed in water to calculate the depth. Survey and commercial vessels have been equipped with these sensors for decades, but have only mapped 10% of the ocean floor [17]. This is because MBES measure a limited swath of ocean, and often vessels need to sail slower for high-resolution mappings. Thus, global coverage using MBES is time and cost-prohibitive. SDB is a method that measures the attenuation of light in a water column, which correlates to the bathymetry. This method has an average error of two meters but is only effective in shallow water, leaving it unable to measure the vast majority of the world's deep oceans.

A team of researchers in India [23] used ML to optimize a local EGM in the Arabian Sea. The researchers found performance increases of 20 meters, and their effort was used to

optimize existing models. ML models can be effective at identifying decision boundaries and relationships that are difficult to identify in human-created models. Specifically, using ML to create models from ocean features provides a possible solution to accurately predicting global bathymetry.

ML is the use of trained models to predict a value. These predicted values can be continuous numbers, known as regression, or discrete labels, known as classification. Multi-class classification is used for predicting many different class labels. For example, if a model is trained to predict bathymetry, then a collection of ocean features and their related bathymetry is the test data. The bathymetry can be placed into classes that represent a range of depth. A class could represent depth values from 2000 to 1850 meters below sea level. This "binning" of bathymetry allows it to be used in a classification model. During training, some of the features will exemplify a particular range of bathymetry. That range receives a corresponding label, and the process will repeat for the rest of the training data. Training the model to correctly identify depth ranges will require features that relate to the bathymetry. Identifying these features is known as feature selection, which can be used to optimize the data that is used during training. Some data will be noise and unnecessary for the final prediction, however feature selection can be used to remove unnecessary training data, improving the final prediction. In this project, a genetic algorithm [10] is used to select optimal features. The genetic algorithm approach for feature selection will select optimum features in a reasonable time frame. After selecting the features, the resulting model can be evaluated for performance and validated against known labels. The more predicted labels that compare accurately to known data, the higher the accuracy of the model. Models, where predictions do not match known data will have poor accuracy. The known data used in this project is bathymetry from existing EGMs. It is important to note that this data is inherently predicted and validated to the best of human knowledge. Each predicted point has an estimated error of 180 meters [17]. Therefore, the data from EGMs is only sufficient for creating theoretical models.

Predicting bathymetry is a complicated problem that involves several unknowns. The vast nature of the earth's oceans contributes to these unknowns. Currently, only ten percent of the earth's oceans have been surveyed, and little is known about the majority of the ocean's ecosystems. Because so little of the earths oceans have been surveyed, if there is to be bathymetry data, then it must be interpolated or predicted in some way. Naturally, it is impossible to measure all of the elements in the ocean. Elements such as oxygen levels, silicate levels, sediment thickness, and sediment type all must be predicted or interpolated. On the other hand, the measurement of satellite altimetry data from the sea surface is readily available. These measurements are accurate, which is why EGMs use gravity as the main feature. However, using estimated features to predict estimated bathymetry could identify new approaches for building viable models.

This thesis is focused on identifying if there is a best fit model, and if there is a way to optimize theoretical predictions. Regression and classification approaches are compared for effectiveness. Models used in this project are all from Python's ScikitLearn library and include: Linear Regression Models, Decision Trees, Random Forest models, Voting ensembles, K Nearest Neighbors classifier, neural networks, and bagging models. A genetic algorithm was introduced for initial feature selection, and a modified grid search was used to identify trends in model performance. Finally, the behaviors of the models were recorded and discussed.

Scikit Learn is an open-source library developed by the Python community [22]. It exposes an intuitive framework for creating ML models. It also provides frameworks for key components of the ML pipeline, such as feature selection and model selection. This framework is implemented for many existing models. New models and components can be implemented that will likewise interface with other pieces of the library. For example, a genetic algorithm for feature selection was implemented in this project using the sklearn Application Programming Interface (API). This component was then able to be used seamlessly with all existing models and other sklearn components.

Grid search is an algorithm for feature selection that exhaustively compares a set of options for the best performing item. The world is split into grids, and the set of options are classification models. The model that performed best in a given grid is selected for predictions in that area. This selection is important for identifying optimal models.

Using classification for bathymetry is not an intuitive idea. Bathymetry is mostly a continuous value that would appear to be ideal for regression models. In reality, the ranged nature of bathymetry makes classification trivial for bathymetry. There is a consistent known minimum and maximum for bathymetry. For example, the maximum depth value for bathymetry is 0 meters, and the minimum is the deepest known depth in the Mariana Trench at 11 kilometers. This range allows for simple labels to be placed on a number of ranges. The number of ranges is arbitrary and can be adjusted to test error. The principal is that selecting a range will infer the maximum error for the prediction. For example, a range of 150 meters infers an error of 150 meters for a point predicted in a class.

2 Literature Review

In this review, different approaches for predicting bathymetry are discussed. The two main methods for collecting bathymetry data are Satellite-Derived Bathymetry (SDB) and Earth Gravitational Model (EGM). There is also a third approach discussed that improves upon the idea of an EGM. Multi-Beam Echo Sounder (MBES) for precise measurements of bathymetry [1] is also covered. The Machine Learning (ML) models used for training are explained in detail, as well.

Machine Learning

Machine Learning can be defined as the process of fitting a model to data with an algorithm [14]. Predictions can then be produced from the models by inputting new data. To validate the predictions, the supplied data's result is known. For example, when a model is trained to predict if an image contains a car, data from the images is extracted and used to train the model. Images with cars are considered positive images. Images without cars are considered negative images. New images that are labeled as positive or negative are given to the model for validation. If the model successfully predicts the labels, it will have high accuracy and be a "valid" model.

The two main types of models in Machine Learning are classification and regression. There are several high-level descriptions of classification and regression models, including supervised learning, unsupervised learning, and reinforcement learning.

Classification models predict a discrete value, for example, whether an image contains a car. These types of models are effective at predicting values that can be grouped into "labeled" data. Data is labeled when it is assigned an output value as a "truth" label. For example, to train a model to detect cars, the model must be supplied with labeled images. The labels are either positive (there is a car in the image) or negative (there is not a car in the image). This labeling describes a "binary" classification, meaning there are two options

from which to choose. Classification models can also be combined to form ensemble models. An ensemble is a combination of weaker predictors to form a strong predictor. Examples of classification models used in this project include: Decision Trees, Naive Bayes, MLP Classifier, Quadratic Discriminant Analysis Classifier, and K Nearest Neighbors Classifier. Examples of ensemble classifiers used in this project include: Random Forest Classifier, Ada boost, Gradient Boosting, Bagging, and Voting.

Regression models predict a continuous value. For example, predicting the value of a home. Essentially, this model represents a mathematical function where the function parameters are instances of new input data. The parameters for the model are predictors for the value of the home. Predictors for a home include square footage, number of rooms, acreage, and school distance. This method is appropriate when the desired result cannot be grouped into discrete "labeled" data. This is because classification is not ideal for predicting an infinite set of continuous values.

Unsupervised learning describes a model that is trained without labels, that is, the training data does not have corresponding truth values [14]. Training a model without "truth" is used for grouping similar values together. This approach is ideal for identifying correlations in data that is otherwise unrelated and unlabeled.

Supervised learning describes a model that is trained with labeled truth data. [14]. Training is guided or supervised by comparing results to labeled truth data. This method of training is effective at fitting accurate models and is used in this project for predicting bathymetry. All models trained in this project use supervised learning. Each model used will be discussed in subsequent sections.

Bias and Variance

The generalized error of a model can be expressed in terms of bias and variance. Bias is the average error of a model for different training sets, while variance describes the sensitivity of the model to data. These two terms are necessary to understand the bias-variance prob-

lem. This problem applies to all forms of supervised learning [5] and describes the indirect relationship of bias and variance. Namely that minimizing bias often increases variance, and minimizing variance often increases bias. Models with high bias will be inaccurate when predicting data not used during training, while models with high variance will overfit [19] the training data by modeling noise in the data as opposed to the desired outputs. Optimally, accurate models should have low bias and variance.

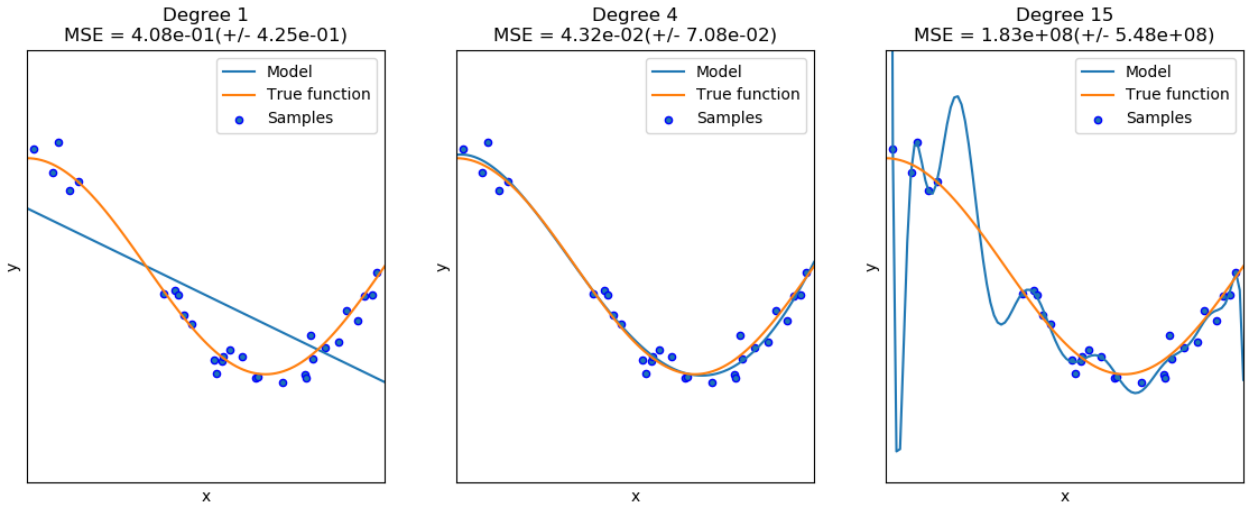

Figure 1: Three plots of the function $f(x) = cos(\frac{3}{2}\pi x)$ with noisy samples along with the graph. A linear regression model is fit to the samples with orders of 1, 4, and 15. The model fit with order 1 does not respond well to changes in the samples and is a great example of high bias. The model fit with order 4 fits the function very well and is an example of balanced bias and variance. The model fit with order 15 fits the samples too well, but fluctuates wildly. It is an excellent example of high variance.

Decision Trees

Decision trees are supervised models used for classification or regression [28]. They are easy to understand and interpret due to their simple decision structure. However, they are susceptible to over-fitting and can create overly complex trees that do not generalize well. Over fitting is where a model will fit too closely to a training set [19], making it unable to make accurate predictions on data outside of what was used in training.

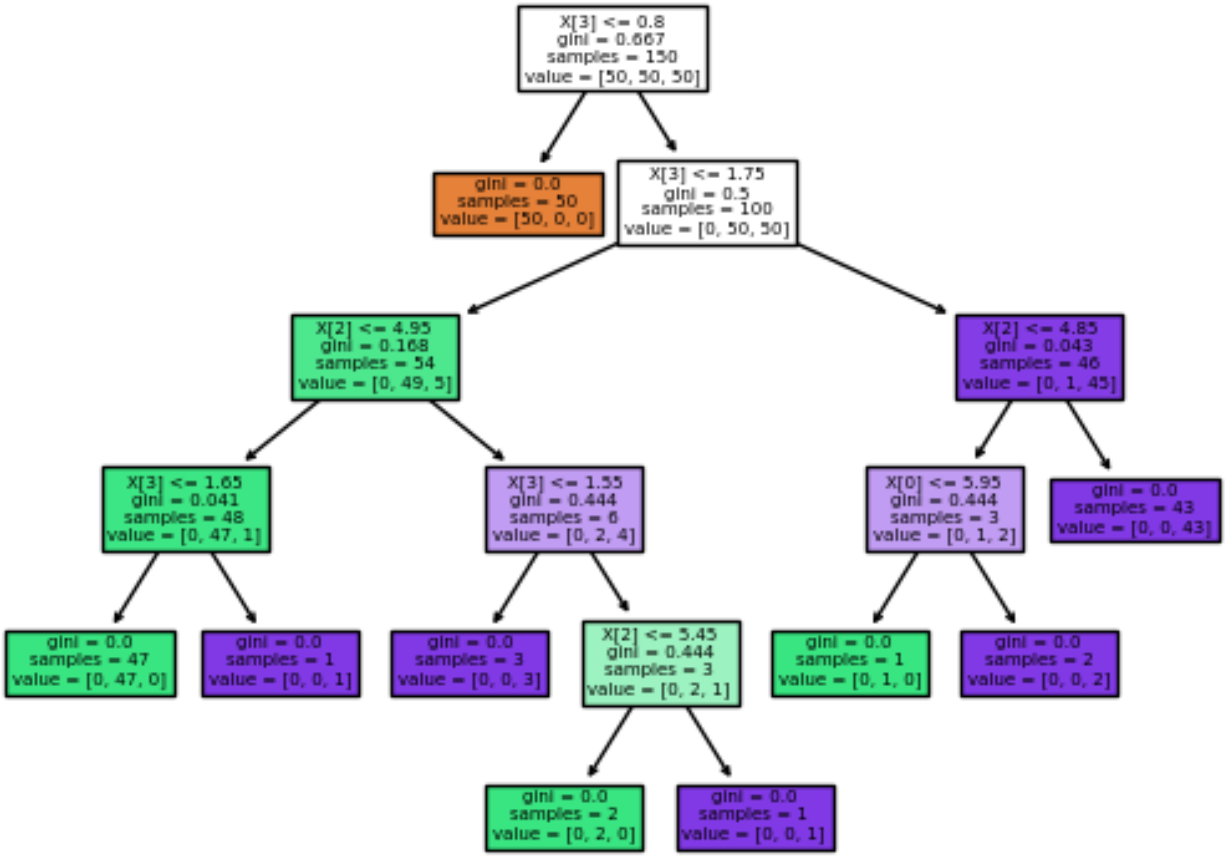

Figure 2: An example of a decision tree. Decision trees work by making binary decisions across a decision boundary. Each leaf node represents a classification label and the binary decisions that selected that label.

The Random Forest Classifier is an ensemble of many decision trees [11]. It fits a number of decision trees on various sub samples of the dataset. This also helps to control over-fitting because each tree is fit to a sub-sample of the dataset. It averages the prediction of each tree to make a more accurate prediction than any single tree.

Boosting

Boosting describes a type of ensemble that is concerned with reducing bias. The idea is to build base estimators sequentially and then use the results of each to train a different estimator with the intention of reducing bias. Adaboost is an example of one such model [7]. Its core principle is the fitting of weak learners on many random samples of the training data, which are then combined with a weighted majority vote. This process is iterated for the training phase.

The gradient boosting classifier is an ensemble of trees similar to a random forest classifier. In gradient boosting, decision trees are used as "weak" learners. They are gradually added to the model in order to reduce the error. This "gradient descent" approach is effective at building a successful classifier. These types of boosting algorithms tend to overfit the model, which can be solved by enforcing tree constraints and random sampling of data.

Averaging

Averaging ensembles operate by aggregating the predictions of many models trained on random subsets of the training data. Introducing randomization into the training will often reduce the variance of the resulting ensemble. Bagging is an example of an averaging ensemble and has several implementations. The bagging ensemble uses a single classifier and fits instances of it to random samples of the training data. The predictions are then aggregated and averaged for a final prediction.

The voting classifier works by using the predictions of a set of conceptually different predictors as votes. The majority vote (hard voting) or averaged vote (soft voting) is selected as the prediction. This classifier is good for combining equally performing models in order to balance out individual weaknesses.

K-Nearest Neighbors

The K-Nearest Neighbors algorithm is a non-parametric algorithm for classification and regression [4]. K-Nearest Neighbors for classification stores the instances of the training data and associated labels. Predictions are made by computing the distance of new instances to the training set. This model is based on the nearest neighbor algorithm and is excellent for its simplicity and domain-independent applications. The nearest neighbors algorithm can be explained using the Post Office problem where a residence needs to be assigned the closest post office [8]. The K-Nearest Neighbors algorithm is a generalization of the Post Office problem. To describe simply, instead of searching for the nearest post office, the algorithm searches for the K-nearest post offices. This can be extended for use in classification. For example, if the K-Nearest Neighbors of an object are a positive class, it is natural to classify the object as positive. See Figure 3 for an illustration.

Multi-Layer Perceptron

The Multi-Layer Perceptron (MLP) is a model that fits a function iteratively through a process called back-propagation. The MLP classifier is a neural network model that is based on the structure of the human brain. It consists of many layers beginning with an input layer. The middle layers are called the "hidden layers" and are each assigned a weight. Back propagation is used during training to adjust the weights in the hidden layers. These adjustments are made in order to reach a more accurate prediction. There can be many hidden layers of different depths. The neural network is a versatile model that benefits from

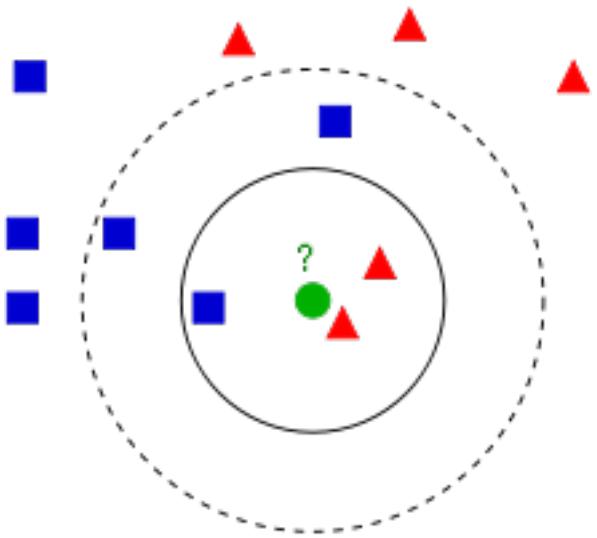

Figure 3: An example of the KNN algorithm for classification. There are 2 classes in this example: Blue squares and Red triangles. The green circle needs to be classified. If K is 3 (the solid line), then the green circle will be classified as a red triangle. However, if K is 5 (the dotted line), then the green circle will be classified as a blue square.

more training data with less risk of overfitting. Training on very large datasets is known as Deep Learning.

Naive Bayes

The Naive Bayes classifier is based on applying the Naive Bayes theorem [12]. The naive assumption of Naive Bayes is the conditional independence between pairs of features given the label. Bayes theorem states the following relationship:

$$P(\theta|\mathbf{D}) = P(\theta)\frac{P(\mathbf{D}|\theta)}{P(\mathbf{D})} \tag{1}$$

In equation 1, θ and $\mathbf{D}$ are events. $P(\theta|\mathbf{D})$ represents the probability of θ occurring given event $\mathbf{D}$ is true, where P($\mathbf{D}$) is the probability of observing event $\mathbf{D}$ alone. Bayes Theorem will produce a probability of an event happening given another event using prior probabilities. For example, say you wanted to predict the probability of both a stock falling and the DOW being down. If you knew the probability of both falling independently, then you could use Bayes theorem to calculate the probability of both happening together.

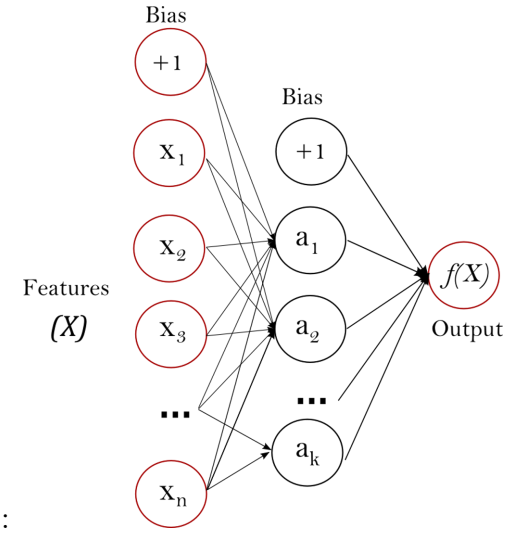

Figure 4: A Multi-Layer Perceptron. The middle or hidden layers are assigned weights that affect the selected class in classification. This model effectively models a function.

Naive Bayes classifiers use an assumption of conditional independence between every pair of features given the value of the class variable. These over-simplified assumptions do not negatively affect the classifiers. In fact, Naive Bayes classifiers work well in many real-world situations because of the conditional assumptions[12]. Famously, it has performed well for document classification and spam filtering.

Regression Models

Regression models are utilized for predicting continuous values. Specifically, regression is used for approximating a function given a set of inputs to yield output. Linear regression is the simplest type and is concerned with fitting a linear combination of all training features. Polynomial regression fits a polynomial combination with the training figures. This can fit a function up to order N with the cost of complexity.

This project utilizes three regression models from Python's sci-kit library. The Naive Bayes[24], logistic regression[24], and svm regression[24] models are utilized for regression.

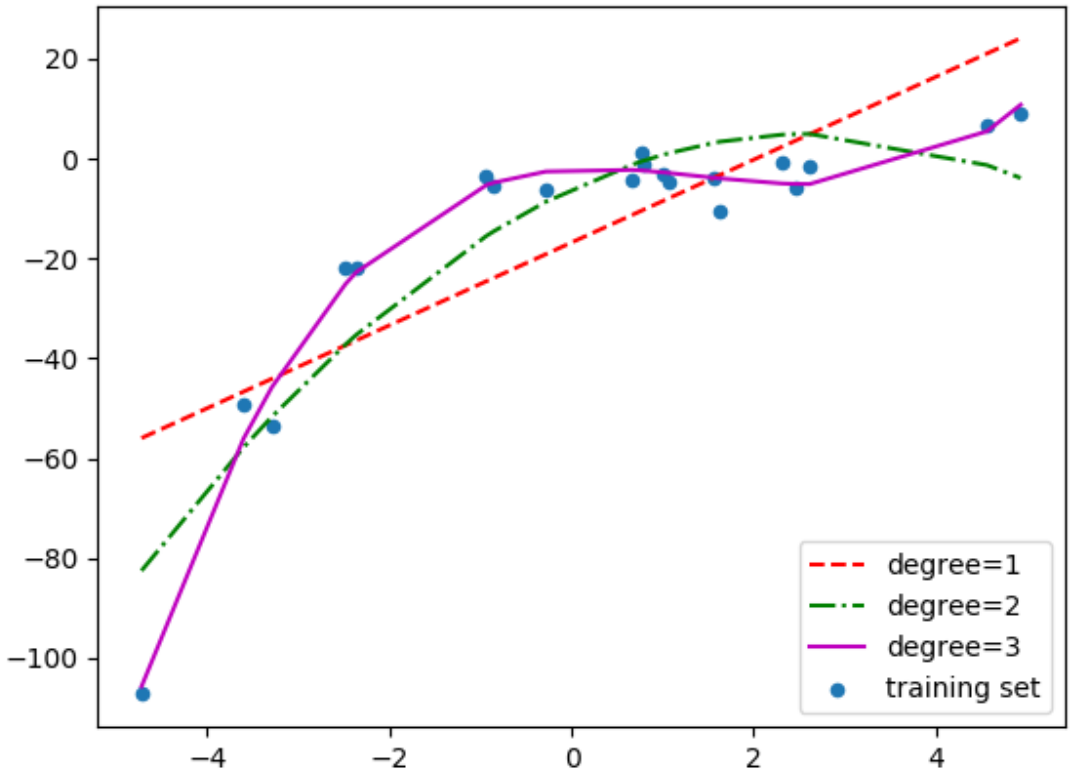

Figure 5: 3 example regression models fit to a set of points. A linear regression (red line) does not fit the data well. As the order rises the line will better fit the data. These higher-order polynomial regression lines come at the cost of complexity and increasing the variance of the function.

Global Bathymetry Data

The oceans of the earth are humanities last frontiers. Ironically, we know very little about the bathymetry of these frontiers [17]. This is due to several factors including but not limited to: measurement techniques, sediment migration, tectonic activity, and the vast size of oceans. This section is concerned with explaining the many measurement techniques for bathymetry and how they are used for creating global bathymetry grids.

Echo Sounders (Multi Beam Echo Sounders)(Single Beam Echo Sounders)

Echo sounders have been mounted on vessels for decades to measure bathymetry accurately. The purpose of this is often for a ship to avoid running aground. This is where a ship strikes the bottom of a body of water. Survey vessels have utilized MBES systems to create reliable bathymetry charts [1]. These charts give an accurate measurement of bathymetry in all water depths. The downside of this method is the cost and time required to map global coverage. A vessel must transport these sensors, and potentially require decades of expensive surveys to gain full global coverage.

There are two main categories of echo sounders: MBES and SBES. MBES are large swath sonar based sensors. They produce several artifacts including: bathymetry, backscatter, and salinity. These sensors are used for a wide array of commercial and research applications. See Figure 6 for an illustration of MBES. Side-Scan sonar is a focused sonar system. They produce "imagery" artifacts and will sense a smaller swath of ocean than MBES sensors. Side-Scan sonar is ideal for detecting objects on the ocean floor due to the image quality of the artifacts. In general, SideScan systems are only used for object detection. Figure 7 shows an illustration of a towed SideScan sensor body.

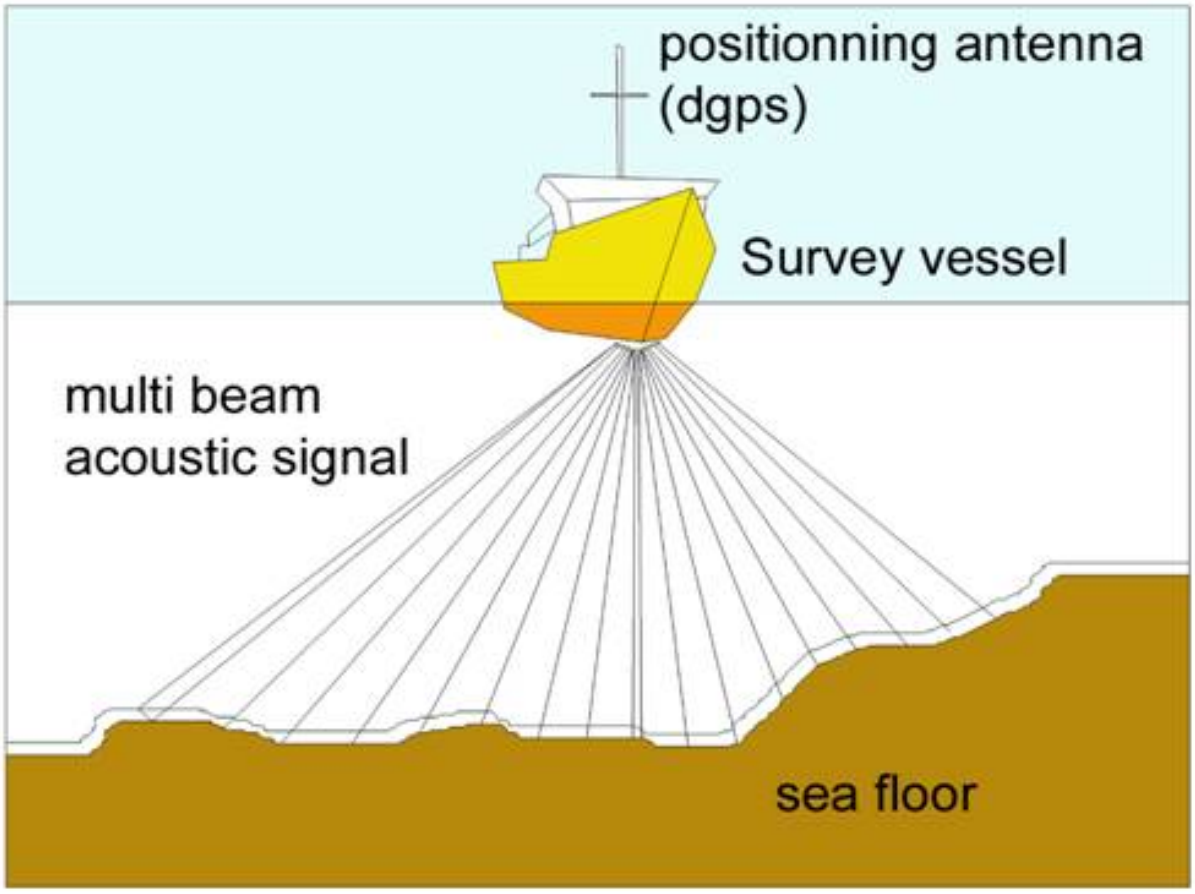

Figure 6: An MBES system illustrated by a hullmounted sensor. The sensor ensonifies the sea floor and listens for the echo of the sound. The time the echo takes to return is used to measure the depth. The illustration was taken from [31].

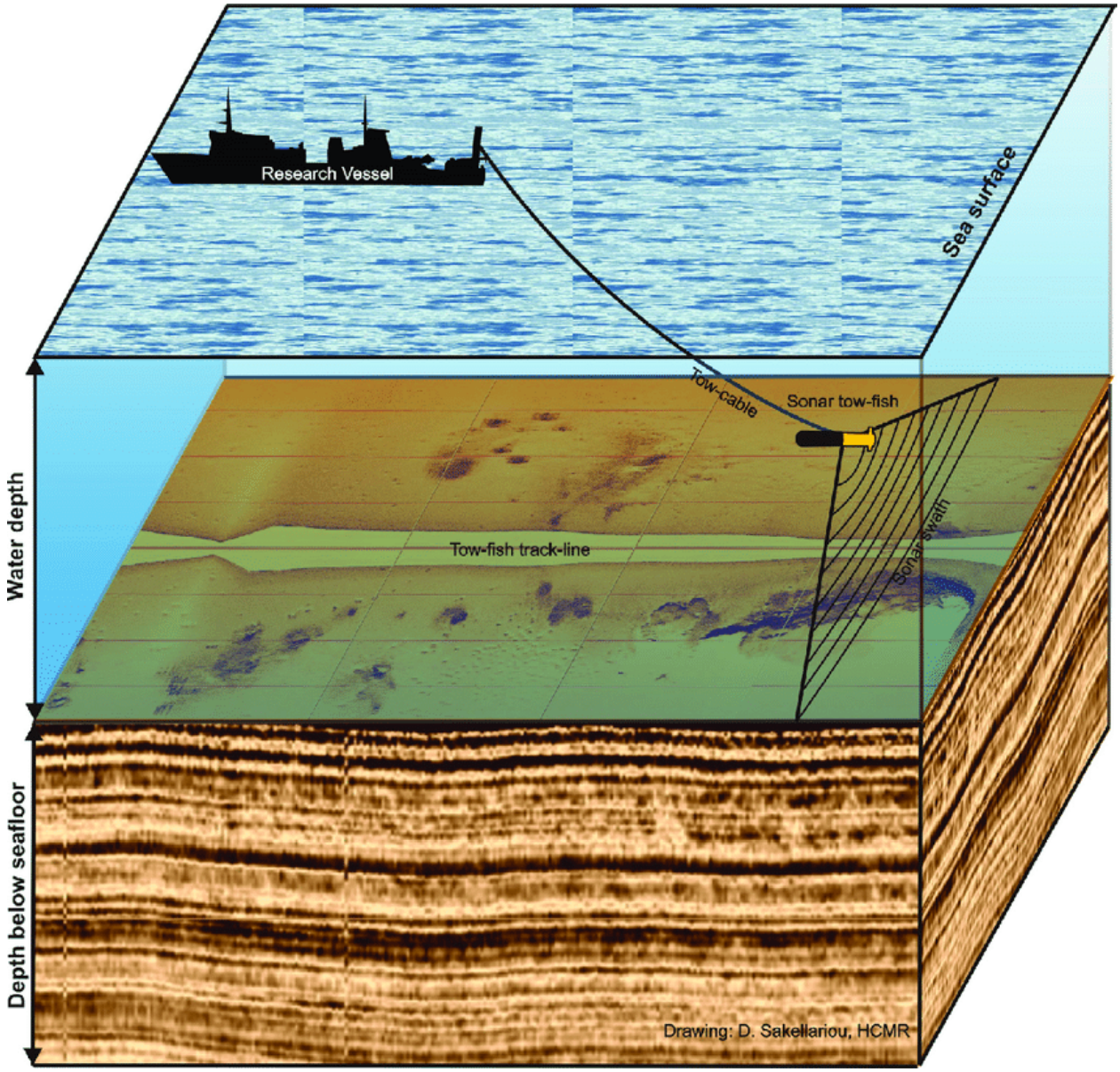

Figure 7: Sidescan sensor used by a survey ship via a tow fish. SideScan sonar is a type of Single Beam Echo Sounder (SBES). A tow fish is a sensor enclosing that is attached to a tow line from the stern of the vessel. This image was taken from [26].

Satellite-Derived Bathymetry (SDB)

SDB is a precise method of predicting coastal bathymetry. This method relies on the phenomena of light passing through a water column at a certain depth described by the Beer-Lambert law [29][27]. Sunlight passes through the water column and is reflected by the sediment at the bottom. Satellites measure the attenuated light that is reflected from the bottom and uses the wavelength to estimate the depth of the column. The technique accounts for atmospheric light absorption, water surface reflection, attenuation through and out of the column, and reflectance from the bottom sediment. Clear waters are the best environment for this method, which has the potential to predict bathymetry with a small RMSE [29].

This method is important for its ability to predict swaths of bathymetry in shallow water where larger vessels can not sail. Large swaths of shallow coastal waters are measured by SDB in a cost and time-effective process. An example where SDB is useful is the marshlands of Louisiana, where water depth is only deep enough for flat-bed vessels. This method also has been used in the scope of national defense for predicting or identifying changes in shallow water bathymetry. These changes can be caused by sediments or man-made objects.

The limitations of the SDB method are based on water depth and clarity. As depth increases, light is unable to pass through the water column to the bottom. The depth that light can penetrate is determined by the characteristics of the water. Clear water will allow for much deeper depths to be predicted, whereas murky, cluttered water limits the maximum depth. Environmental characteristics, such as sediment composition and weather affect the clarity of water [27].

Current SDB models can predict bathymetry with a Root Mean Square Error (RMSE) of less than 2.5 meters at a maximum depth of 50 meters. Work performed by [27] has improved the depth by using blue light-sensing techniques. Work performed by [29]

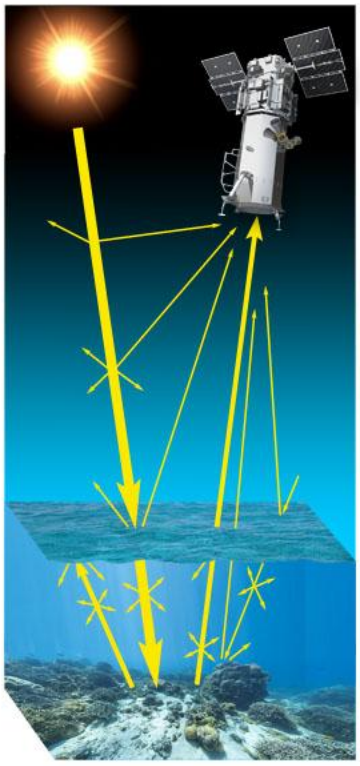

Figure 8: SDB. Sunlight penetrates the water column and reflects back into space where it is captured by a satellite.

improved the accuracy by using advanced regression models when measuring the wavelengths. SDB models are ideal for coastal areas with high water visibility.

Aggregated Earth Gravitational Models (EGM)

Work performed by Smith and Sandwell [6, 9] pioneered the use of Earth Gravitational Models (EGM) for predicting bathymetry. Dixon et al. demonstrated a correlation between depth and sea surface gravimetry by comparing known bathymetry and known geoid heights with satellite measured altimetry [2]. Using this correlation, Smith and Sandwell developed a model for global predictions [6]. Their work identified the wavelength bands at which this relation holds true. They identified the areas that could be predicted with this relationship and used sparse ship soundings to fill in the gaps of their predictions. Areas with large seamounts found the correlation to be strongest, while areas of flat sediments found the correlation to be weakest. They named this technique the "Inverse Nettleton Procedure" and used a simple linear regression model to exploit this correlation and improve existing aggregated datasets.

$$b(x) = D(x) + s(x)g(x) \tag{2}$$

Equation 2 represents the model defined by [6], where **b** represents the predicted bathymetry. **D** represents a function of regional estimated depth via satellite. **s** representing a regional scaling factor based on known sediments. Finally, **g** represents inferred gravity from satellite imaging.

Smith and Sandwell aggregated their predicted values from their EGM with ship-based MBES sonar data. The sonar data is sparse, but provides accurate readings of the ocean's bathymetry. This aggregation yielded global coverage up to 81 degrees latitude. The aggregation yielded prediction accuracy within $\tilde{1}00$ meters in coastal waters and $\tilde{2}00$ meters in the global ocean space.

This work was later improved in [17]. By increasing the number of ship soundings and increasing the aggregation sources. Ten external datasets were aggregated for the SRTM30 grid. Agencies in these sources include the Naval Oceanographic Office (NAVO), General Bathymetric Chart of the Oceans (GEBCO), National Oceanic and Atmospheric Administration (NOAA), National Geospatial Agency (NGA), and the Japan Agency for MarineEarth Science and Technology (JAMSTEC). These datasets include high-resolution coastal bathymetry from around the world and greatly increased global accuracy

The limitations of aggregated EGMs are based on "correlation uncertainty" and unknown ocean features. Sediment density drives the correlation between sea surface gravity and bathymetry. A dense geoid will generate greater sea surface gravity than a less dense geoid. Flat sea floors with a less dense composition can appear lower than normal. This is all controlled by the scaling factor described in [6] and shown in equation 2. There is not, however, an optimal scaling factor for the entire world. Identifying an optimal scaling factor for an area will require knowledge of the sediment type on a global scale.

Machine Learning Optimized EGM

The aggregated EGM is a physics-based model that relies on the relationship between gravity and bathymetry to be directly correlated. This correlation is often non-trivial to define due to environmental factors. The nondeterministic behavior is compensated by the scaling factor detailed in section 3.5. Attempts to optimize this scaling factor have been made, and ML has shown much promise in this regard [23].

The work performed by [23] used ML to optimize their gravitational regression models. Instead of identifying an optimal scaling factor deterministically, they used an Artificial Neural Network (ANN) to optimize the scaling factor. This was done using precise MBES data as truth data and satellite altimeter data. This method was tested on a localized swath of the Arabian Sea.

This optimization improved upon the current physics models. Their model could predict bathymetry within an RMSE of $\tilde{1}29$ meters for a section of flat sea-floor. Geoids resulted in an RMSE of $\tilde{1}79$ meters. Both of these results are globally similar to aggregated models while providing increased accuracy in their localized training area of the Arabian Sea.

3 Methodology

The following chapter describes the data used in this project and how it was used for training. Bathymetry data was extracted from the Earth TOPOgraphy (ETOPO)2v2 and used as truth value to train against a set of ocean features. These ocean features were selected from a large set of aggregated features. The selection was driven by a genetic algorithm that was used for speed as opposed to accuracy. The trained models are then used to make predictions for the spatial resolution that was used in training.

Bathymetry Data

Data derived from predicted EGM datasets represent the actual value of bathymetry at that point. For example, predicted bathymetry datasets have geospatial coverage. These coverages vary in size by the resolution of the imagery. Higher resolution imagery will have smaller coverages, and lower resolution will have more extensive coverages. The predicted values used in this project represent an average of bathymetry across coverage. These predicted datasets are used as the truth values for training, but it is important to note that they are, themselves, predicted values. Using this data is not intended to build accurate predictors, but to show that an accurate predictor can be built.

The ETOPO dataset is used in this project for bathymetry data. This dataset is an aggregation of sparse MBES ship soundings and predicted bathymetry from an EGM. It is an updated version of the original ETOPO2 dataset and was chosen because of the two-minute resolution it offers. ETOPO was aggregated by the National Geophysical Data Center (NGDC), which is a department of NOAA.

Land topography is included in the ETOPO dataset. This proved to be problematic in creating accurate bathymetry predictions. Therefore, a mask was created to remove the land topography in all training datasets. This is applied to the data before training to ensure that land data is not used in training.

Classification requires discrete class labels to predict. To accommodate this requirement, the ETOPO dataset was binned into discrete classes. This binning was performed at 150-meter intervals. This partitioning scheme was chosen to compare to the results from a similar work [23].

Training Data

All feature data was aggregated from ocean and earth studies. This data was then normalized and formatted for this project. This aggregation was performed to gather a large set of testing samples regardless of the data's relevance. For example, crust age may not be obviously useful for predicting bathymetry, but if it is useful, then feature selection will identify it as such. This approach is fundamentally different from building a physics model. For example, the mass of a geoid causes a vertical gravity gradient, which creates ocean swells that can be measured by satellite altimetry. Larger geoids will naturally affect the bathymetry of the sea floor. This relationship is obvious and correlated in a measurable way. On the other hand, machine learning can potentially identify relations in the data that otherwise will be difficult for a human to identify and model.

All features and their origin datasets can be seen in Table 1. Absent data points were either interpolated or filled with default values.

Figure 9 shows a plot of estimated fish biomass and bathymetry. It can be conjectured that this relationship is more correlation than causality. For example, biomass increases are not caused by shallow depths. The shallow depth has more available light, which allows for vegetation and energy supplies for more species, which could explain the relationship shown in this figure.

Figure 10 shows a graph of crust density and bathymetry. This relationship may be more of causality than correlation. The denser crust is caused by many different factors that are separate from bathymetry. It is possible that deeper water columns and the resulting weight contributed, but it can not be used to describe the correlation of the variables.

In general, the features used in this project were a collection of potential predictors. Features such as estimated oxygen, nitrogen, and salinity make sense for being related to bathymetry. Other features, such as crust density may not naturally be explained. The feature selection was used to identify the best performing features. Future studies in the relationship between these features and how they benefit predicting bathymetry will be necessary.

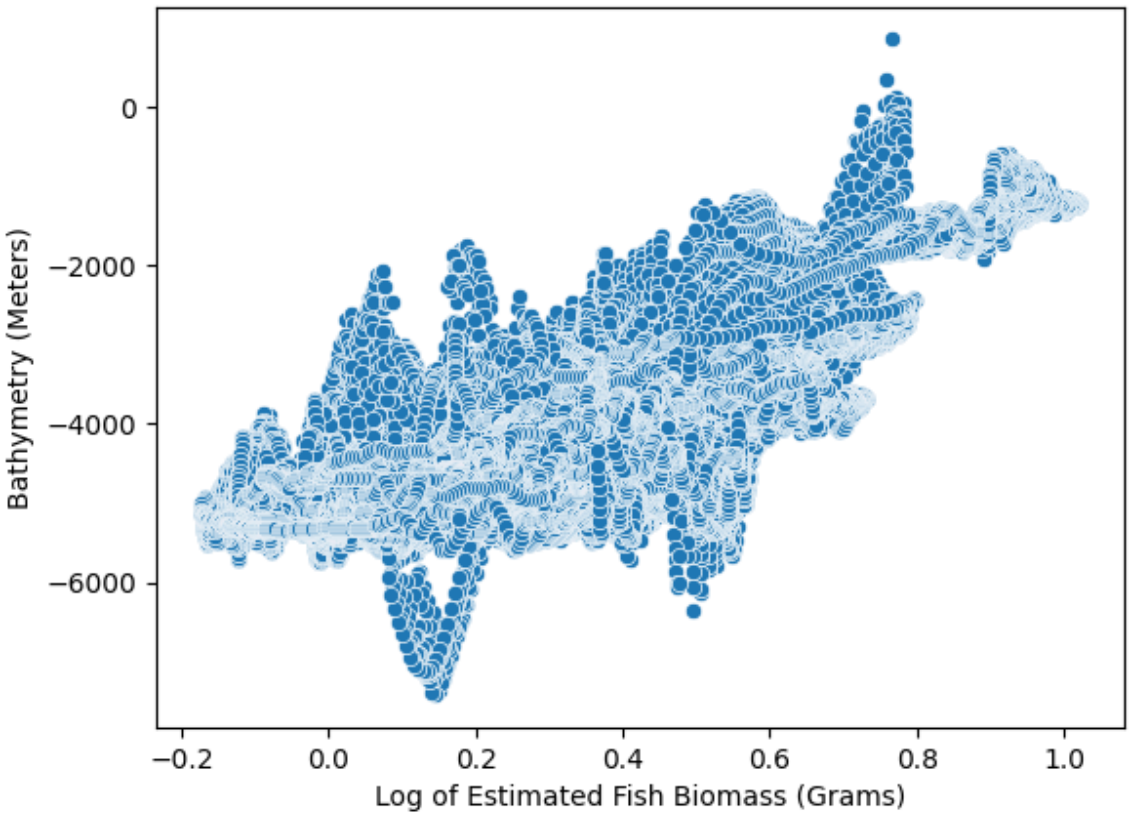

Figure 9: Graph of Bathymetry and Estimated Fish BIOMASS. Bathymetry is measured in meters and Fish Biomass is measured

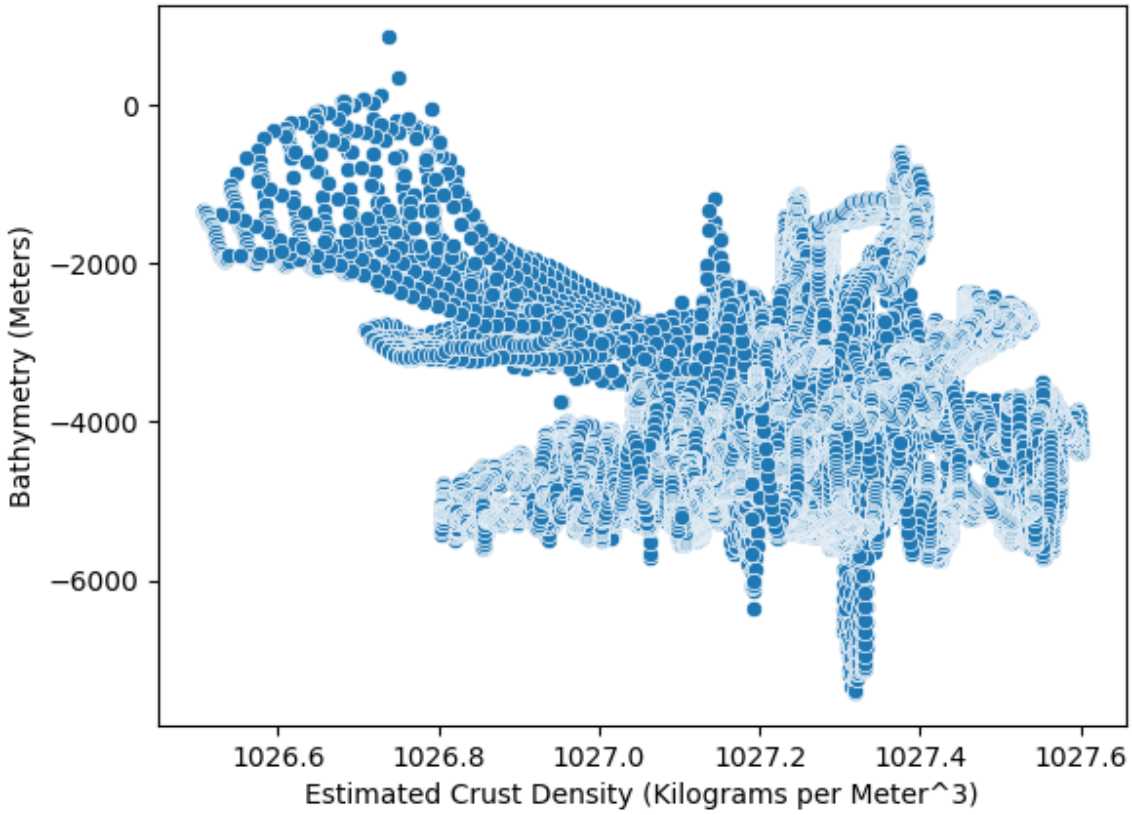

Figure 10: Graph of Bathymetry and Estimated Crust Density. Bathymetry is measured in meters, and Crust Density is measured in milligrams per squared centimeter.

Table 1: List of Ocean Features used in Models for this project.

Feature	Origin Study
Mantle Density	CRUST1 [25]
LAND One Hot	ETOPO [3]
Crust Thickness	CRUST1 [25]
Low, Mid, High Crust Density	CRUST1 [25]
Estimated Current East, North, Mag	HYCOM [18]
Sea Nitrate, Phosphate, Salinity Measurements	NASA Studies [30, 13]
Sea Temperature, Silicate Measurements	NASA Studies
Sediment Thickness	CRUST1 [25]
BioMass Features	CRUST1 [21]
Geoid Features	EGM [16]
Wave height, period	WAVEWATCH [15]

Feature Selection

Feature selection was used to identify the most relevant features for classification. This important step in the ML pipeline removes noise from irrelevant data. This work used a genetic algorithm approach for feature selection [10]. Other approaches that were considered included a grid search, dimensional analysis, and simple variable correlations. These approaches were found to either take too long or simply not offer enough improvement to the model. The genetic algorithm approach gave relatively quick model improvements with little effort. See Figure 11 for an illustration of a generic genetic algorithm.

Using a genetic algorithm for feature selection is a simple application of the original process. The initialized population is a set of random binary strings. Each string has a character length equal to the number of features in our feature space. The binary characters represent whether a feature is active or inactive. Essentially, these strings represent a set of features to use in training a model. The fitness of that string is represented by the resulting model's accuracy. Selection is performed by choosing the most accurate models and their characteristics then passing those onto the next generation. A simple crossover mutation of

the strings is used along with a modest 5% mutation rate. Upon termination, the resulting fittest string is used as the selected features.

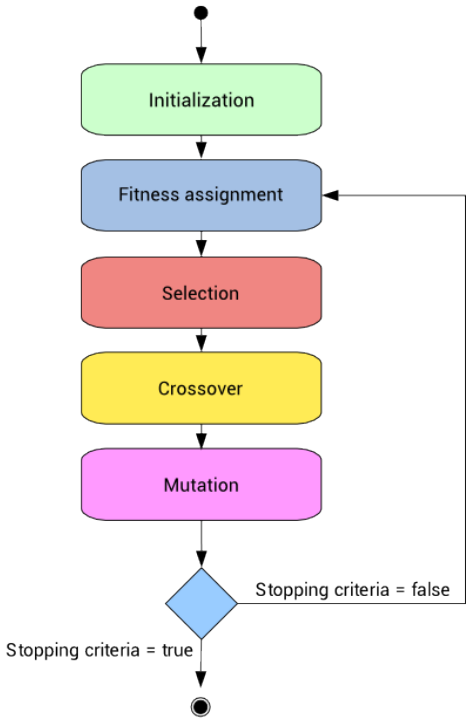

Figure 11: State Diagram of Generic Genetic Algorithm. The algorithm begins at an initialization state where the population is created. It then assigns a fitness attribute to each member of the population. This is the Fitness Assignment State. Then in the Selection State, a selection is performed where the highest-ranking members are selected for crossover and mutation. These states are designed to replenish the population. These actions are performed in the Crossover and Mutation States, respectively. Finally, a stopping criterion is evaluated. If the population meets this criterion, the algorithm exits; else, it goes to the Fitness Assignment State and repeats the previous steps.

Feature selection was used to remove noise from the training data. This down sizing happened on a global scale. It is possible that this may affect localized predictions where a feature is a strong predictor. This was not tested in this work, but it is an interesting question. Identifying locally optimum features could lead to better models.

Data Representation

Representing the topography of the earth can be done using a grid. Naturally, mapping a large circular sphere to a flat grid is not a direct conversion. Latitude and Longitude represent the grid lines for the earth. The data in the grid is a representation of the average value across an area. This grid representation allows for ease of reading both by computers and humans.

The spatial resolution of a grid defines its coverage. It can be described as the height and width of a grid. This height and width are not spatially constant. For example, a cell at the equator is larger and covers more physical area than a cell at the poles. This format is preferred because it represents the data in a consistent and structured manner.

All data in this project has been organized into two-minute bathymetry grids. A two-minute bathymetry grid has a spatial resolution of 0.034 degrees per cell, which is approximately 3 kilometers of spatial coverage. The grids have a column length of 5400 and a row width of 10800. This resolution was chosen for experiments to conserve memory and time. Larger grids have an exponentially larger memory and computational footprint. I used the ETOPO2v2 [3] dataset as the source of the two-minute bathymetry grid. Finer resolution datasets exist, such as the SRMT30 [17] at 30 second resolution, however, the two-minute resolution offered a good balance of memory, accuracy, and computational costs.

Metrics

Metrics are useful for evaluating a model and determining its usefulness. The metrics used in this paper were chosen with this goal in mind. RMSE, R2, F1-score, and Balanced Accuracy are the metrics used in this paper.

RMSE stands for Root Mean Square Error. It is the square of average squared error, and is ideal for aggregating the magnitudes of error. It is given by the following equation. In this equation, d_i represents a true known value and f_i represents a predicted

value.

$$RMSE = \sqrt{\frac{1}{n}\Sigma_{i=1}^{n}\big(d_i - f_i)^2} \tag{3}$$

R^2 stands for "R Squared". In statistics, this is the coefficient of determination, it is defined as the proportion of the variance in the dependent variable that is predictable from the independent variables. Having a high R squared score will denote that the features of a model are good predictors. It is given by the following equation. In this equation, y_i represents a known value from a dataset that is associated with a predicted value f_i and μ represents the mean of the series of known values $[y_1...y_n]$.

$$R^2 = 1 - \frac{\Sigma_i (y_i - f_i)^2}{\Sigma_i (y_i - \mu)^2} \tag{4}$$

F1-Score represents a harmonic mean of a model's precision and recall. Precision is the measurement of how good a model is at avoiding false positives. It is known as the true positive rate. The recall is a measurement of how good a model is at avoiding false negatives. The equations are given below.

$$precision = \frac{TruePositives}{TruePositives + FalsePositives} \tag{5}$$

$$recall = \frac{TruePositives}{TruePositives + FalseNegatives} \tag{6}$$

$$F1\ Score = 2 * \frac{precision * recall}{precision + recall} \tag{7}$$

Balanced Accuracy gives an indication of accuracy for datasets that are not balanced. A dataset that is not balanced means that one label has significantly more or fewer samples than another. This is especially useful for this work because the training data is very imbalanced. Imagine a classification problem where there are 100 samples. 95 of those samples are positive and 5 are negative. If a classifier predicts all 100 as positive, then it will have a 95% accuracy if you just compare the correctly classified class against the whole.

However, balanced accuracy will give a score of 50%, which gives a better indication of the success of that classifier. The equation is given below.

$$Balanced\ Accuracy = \frac{TruePositiveRate + TrueNegativeRate}{2} \tag{8}$$

Learning Methodology

Supervised regression, classification models, and a novel ensemble were trained for this research. The training was performed using previously predicted bathymetry as the truth data. Potential predictors in the form of Ocean data were aggregated and used as training features. Predictors that performed well were selected by a genetic algorithm.

Regression Methodology

Regression ML models were fit in order to compare to existing physics models. Three models were fit to the data: an SVM regression model, a Naive Bayes regression model, and a simple linear regression model. These models were trained against a reduced set of data shown in Figure 12, and validated against the world-wide data not used in the training phase. Each model was fit against selected features from the Genetic Algorithm feature selector.

Classification Methodology

To facilitate classification, trained models are needed to predict discrete values instead of continuous values. This conversion was executed by mapping the bathymetry values into discrete classes. The conversion proved to be trivial because of the ordered nature of bathymetry. Classification models are simpler and easier to fit than regression models. Ideally, the decision surfaces will benefit from the conversion and yield better results. This makes it difficult to compare directly to the error reported in existing physics models. A set of metrics, in-

cluding precision, recall, F1-score, and Balanced Accuracy were used to analyze the viability of the models.

The ordinal classes were binned on an interval of 150 meters. Essentially, this gives a true positive error of approximately 150 meters. This was done to easily compare accuracy to the model in [23]. Validation was performed with 10-fold cross-validation using Balanced Accuracy as the scoring function. The following models were imported from the Sklearn library and trained: Random Forest Classifier, Quadratic Discriminant Classifier, AdaBoost Classifier, Gradient Boosting Classifier, Decision Tree Classifier, Artificial Neural Network Classifier, Voting Classifier, KNN Classifier, Bagging Classifier, NaiveBayes Classifier.

Model Selection Methodology

For optimization a grid optimization methodology was developed. Essentially, grid optimization uses a decision function to select the best predictor. A cache of models was used for this decision function, specifically, the cache is a map of a model to a geospatial area. The methodology is that specific models will be better at predicting specific geospatial areas. This cache was implemented by comparing the performance of a set of models for predicting across the globe. For sake of time and computing resources, the world was split into **N** coverages. These coverages represent an area of the earth and sufficed for the experiment. Each model was trained using data contained by a radius surrounding the coverage. Using 10-fold cross validation each model was scored and then compared. The best performing model was then placed into the cache. For predicting globally, each model was trained and validated against the entire dataset.

Model selection using this grid optimization methodology allowed the best predictor to be used at all times. Figure 17 shows that model accuracy varied greatly by region. Showing evidence that there does not exist a best fit model for predicting bathymetry across the earth. For example, consider a model that is excellent at predicting shallow bathymetry. This model's performance could be explained by how it interprets data and models it. Specif-

ically, it may leverage the predictors in the data well. However, this same model that works well in shallow water may be awful in deep water. Whereas, another model may be excellent at predicting bathymetry in deep water scenarios. This methodology solved this problem by always using the best predictor for an area.

4 Results

This section contains the results for each model observed by this work, including results for the regression, classification, and novel grid optimized model. Each section includes figures representing the metric scores.

Regression Results

The R^2 score for each model can be found in Figure 13. The RMSE of each model can be found in Figure 14.

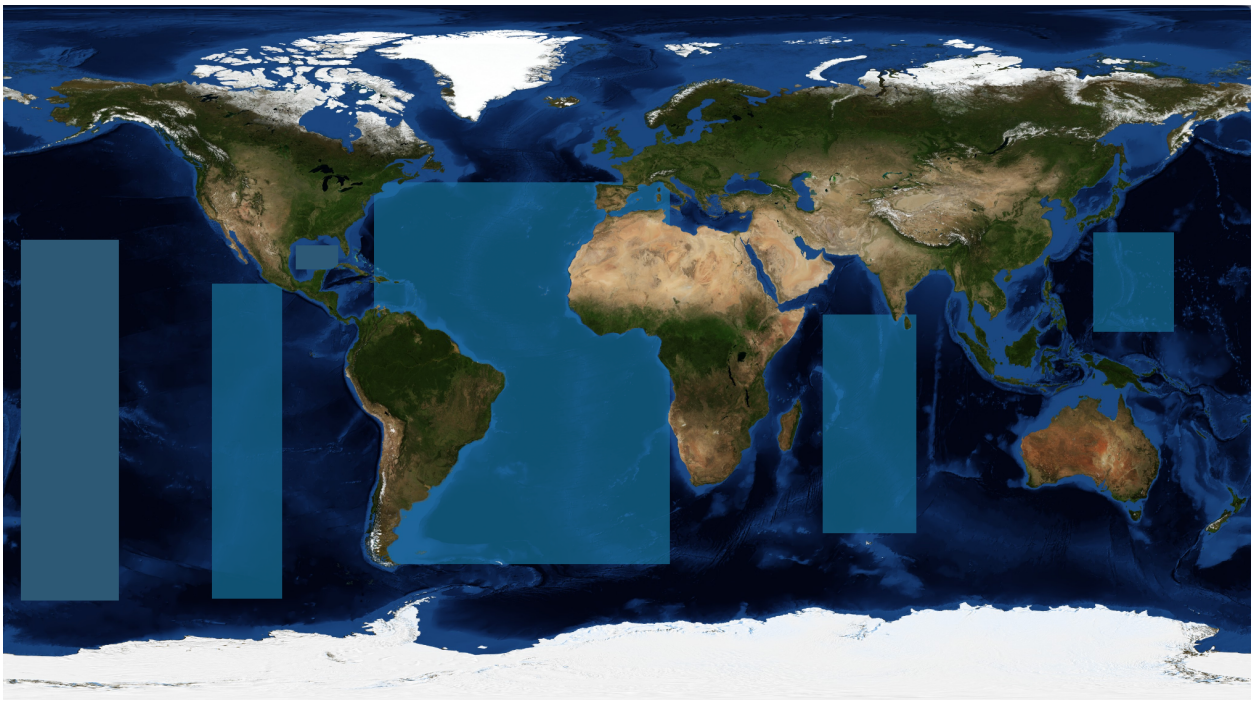

Figure 12: Initial training sets for regression. Testing was performed against the rest of the world.

Classification Results

The F1 results for each model can be found in Figure 15. The Balanced Accuracy for each model can be found in Figure 16.

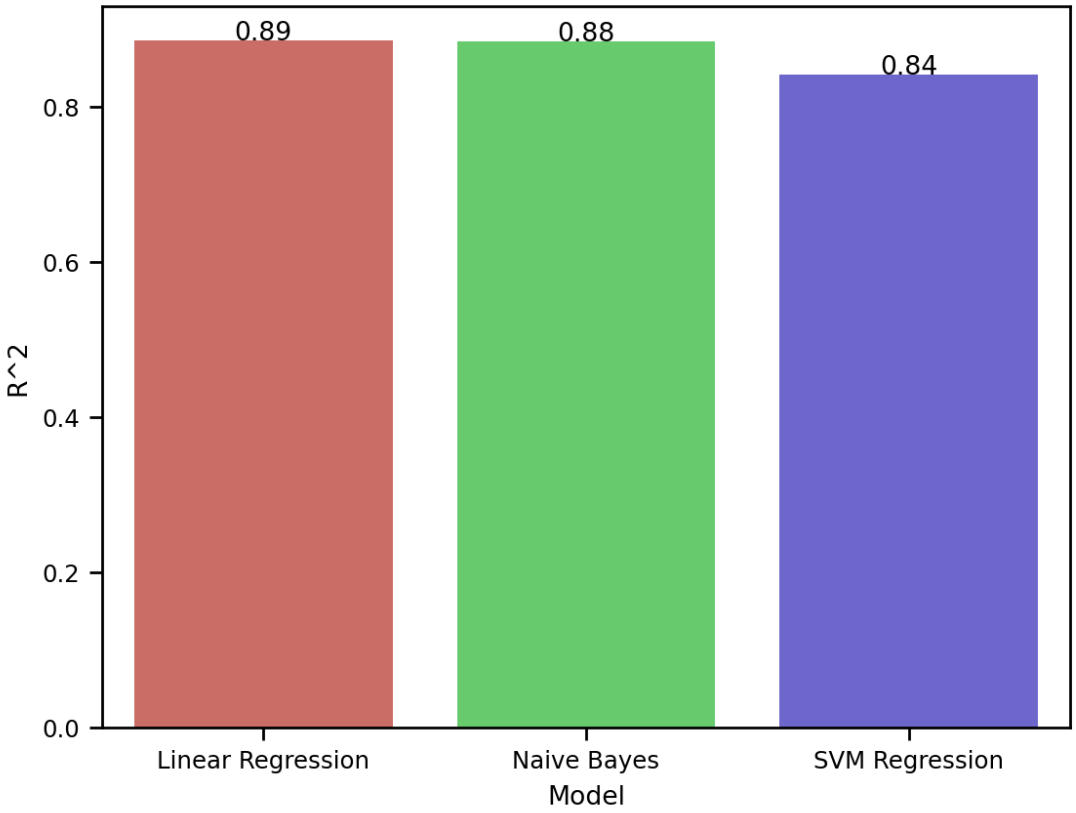

Figure 13: R^2 score for each model. Higher values represent better performing models.

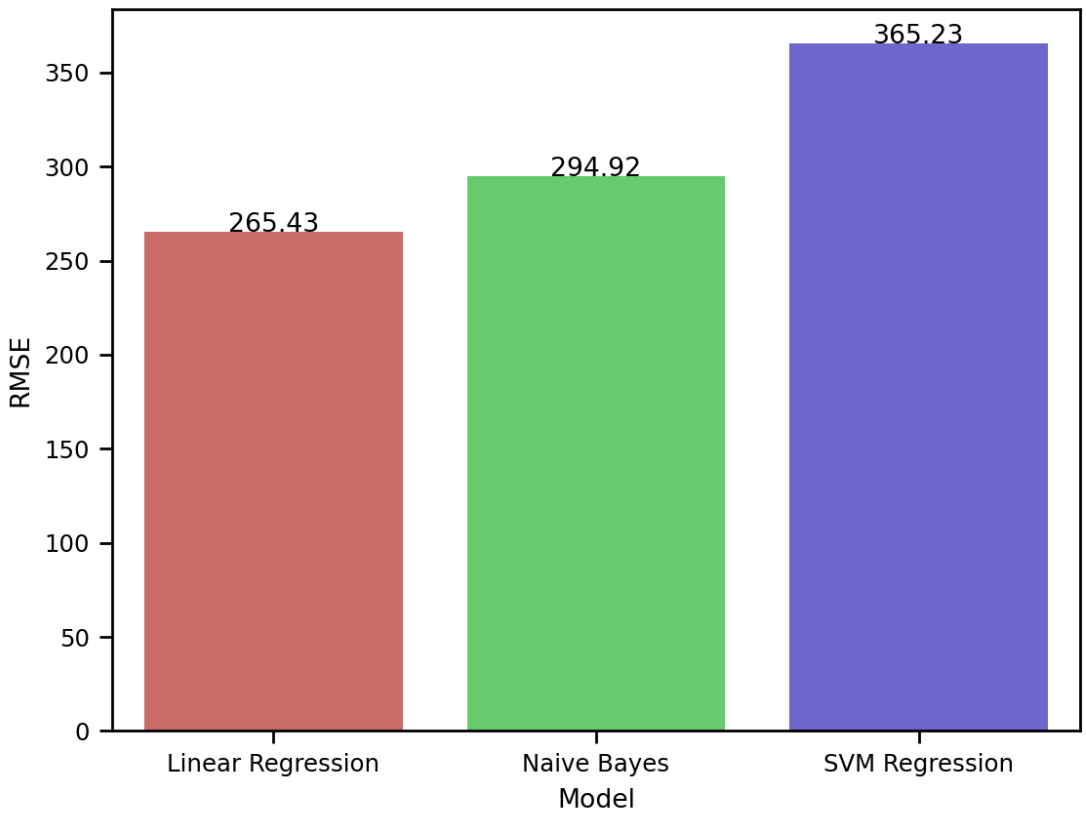

Figure 14: RMSE for each model. Lower values represent lower error for a model.

The Decision Tree classifier performed significantly worse than the other models. The 47% balanced accuracy is not usable for predictions. Potentially, parameter tuning and feature selection could improve this model.

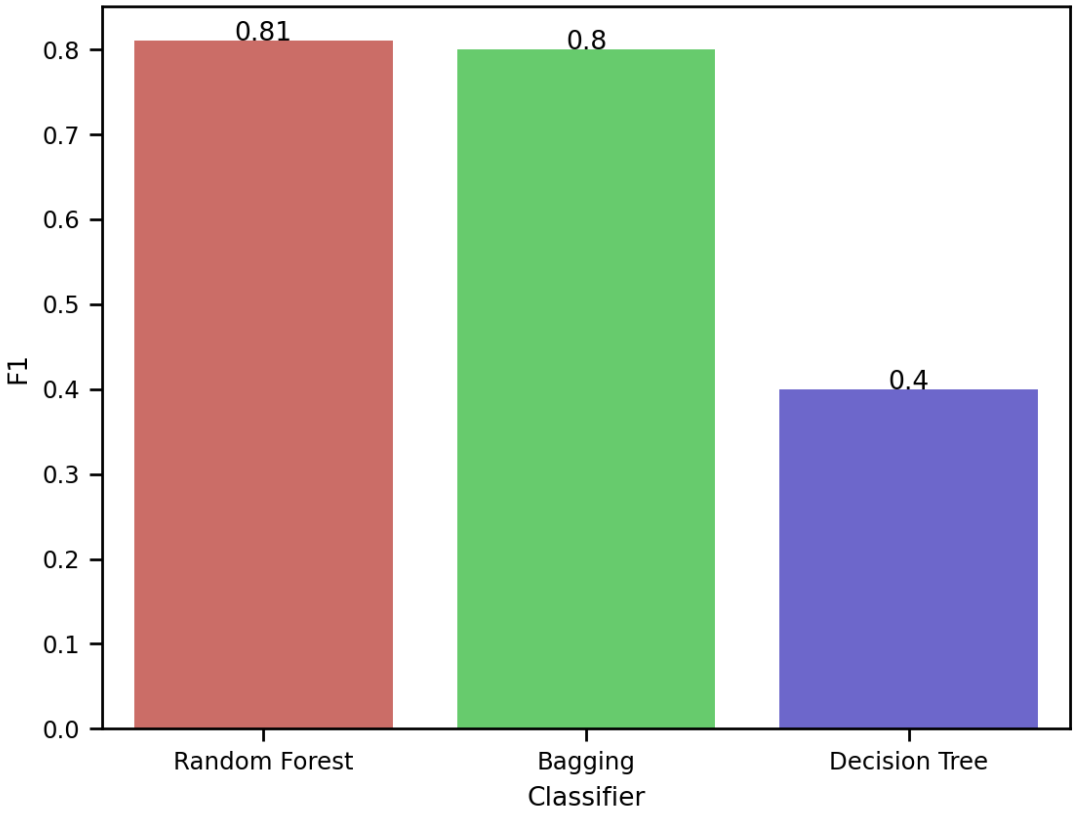

Figure 15: F1 score for each classifier. Higher values represent better precision and recall. From this, we see the Decision Tree does not make useful predictions.

Grid-Optimization Results

The Grid Optimized Classifier used the results from Figure 17 as a decision function to select an *optimum* model. Simply, the classifier checks the location of the prediction for its corresponding grid in Figure 17. The model that performed best in that coverage is then used for classification. This optimum model selection improved the prediction accuracy of the model by several percent. See Figure 19 for the results of the model.

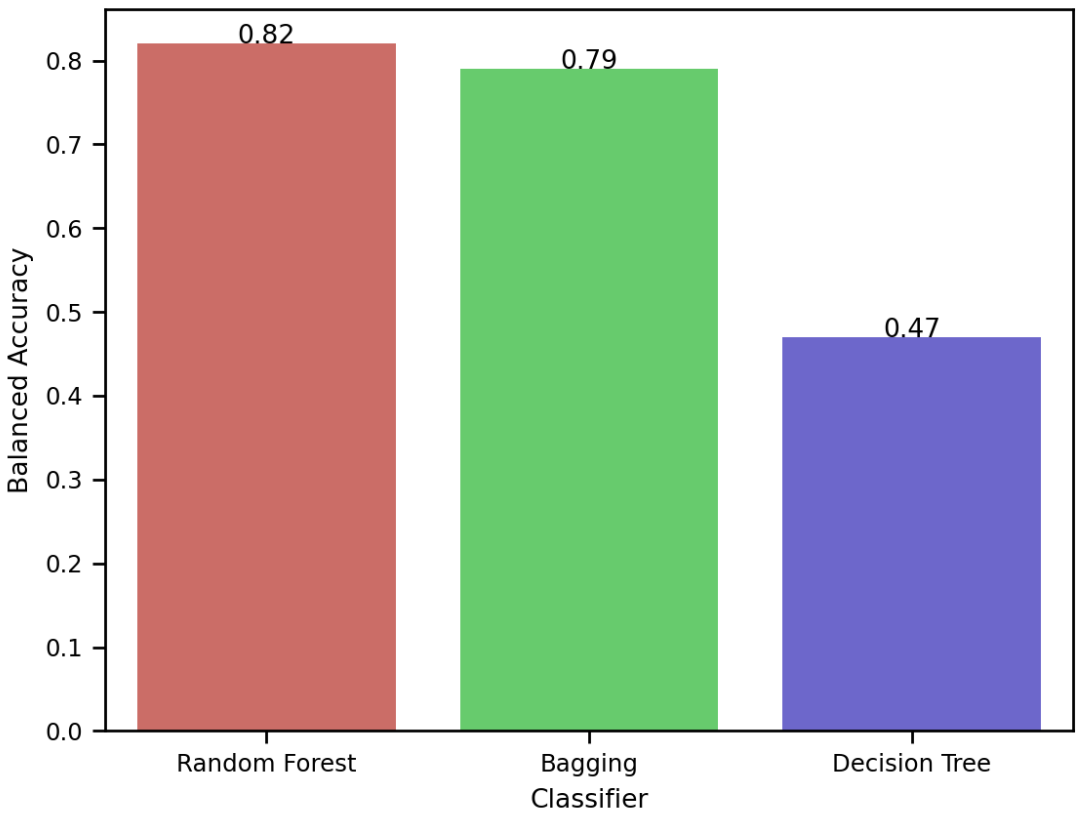

Figure 16: Balanced accuracy for each classifier. Higher values represent higher accuracy.

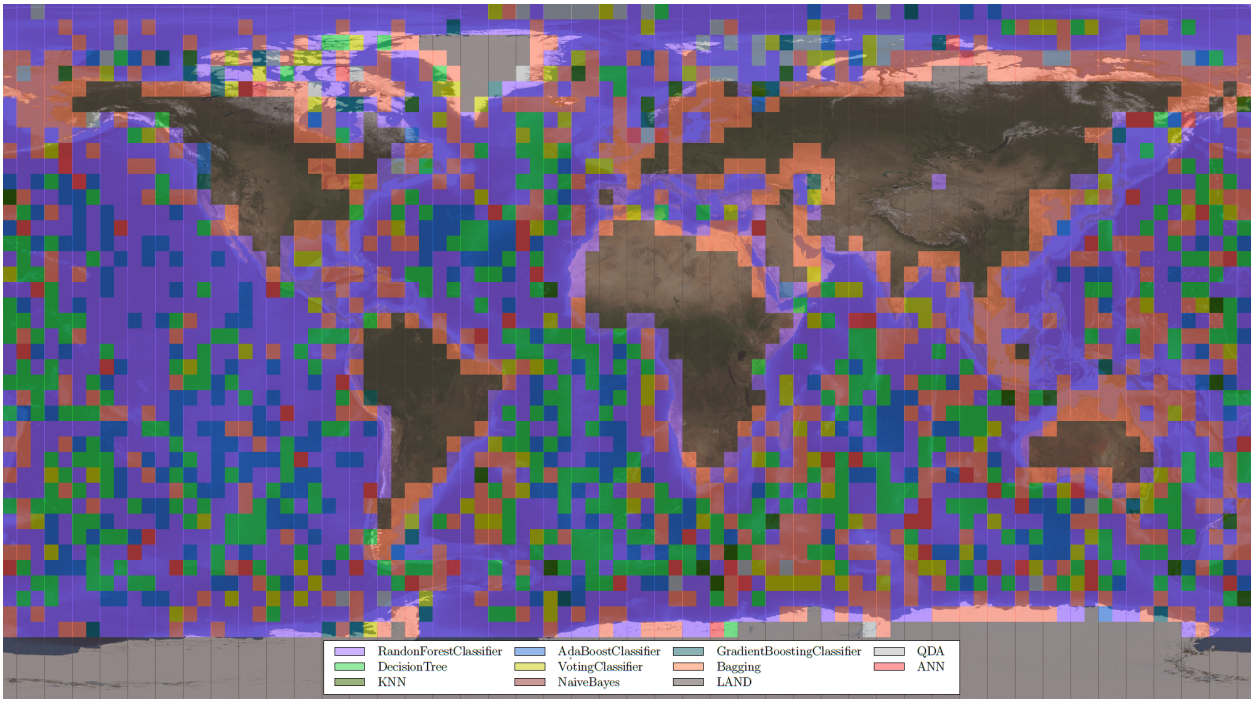

Figure 17: World Coverages and Successful Models. Each square represents a coverage. The shaded color represents the model that was most successful in that coverage.

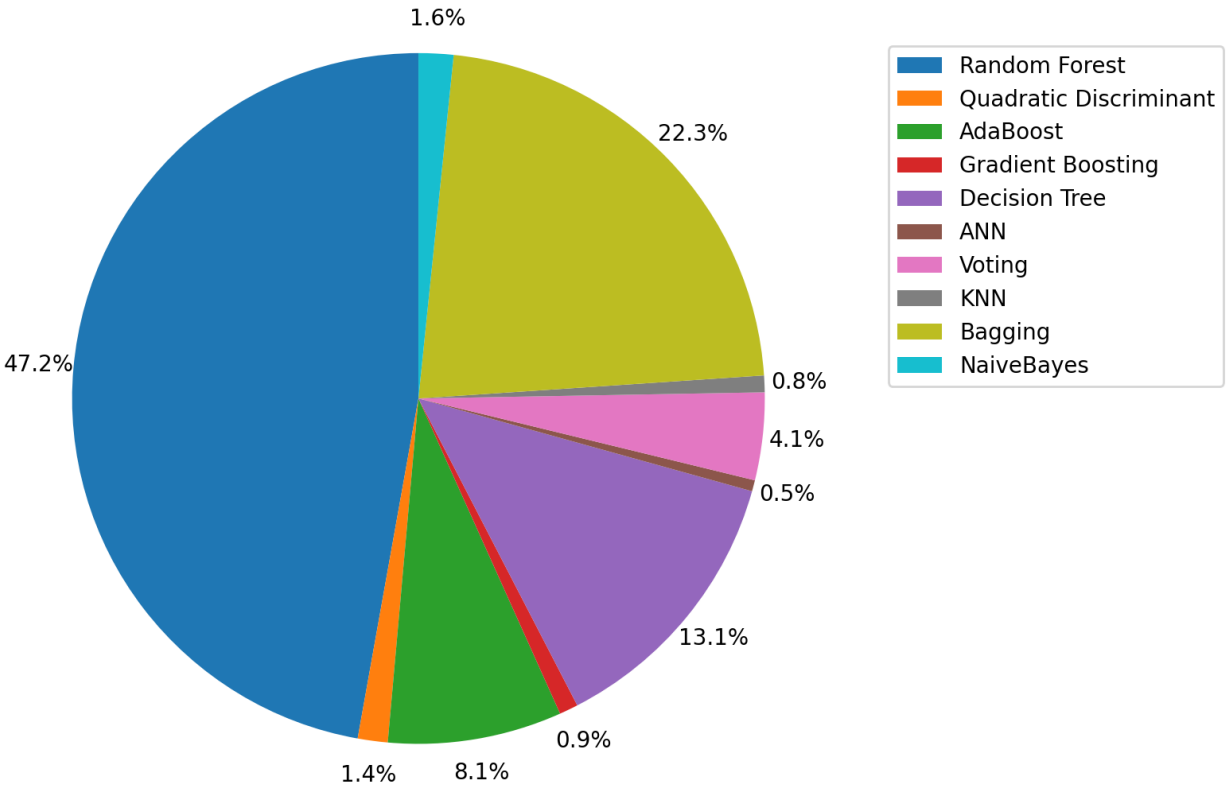

Figure 18: Percentage of coverages where a model was "best fit".

Figure 17 illustrates the coverages where each model performed best. Figure 18 shows the percentage that each model was a best fit. The random forest classifier was the best fit model for a large portion of the oceans. On the other hand, the Bagging classifier consistently performed well along the coast lines. The reasons why these classifiers may have performed so well in those areas will be discussed/explained in Section 6.

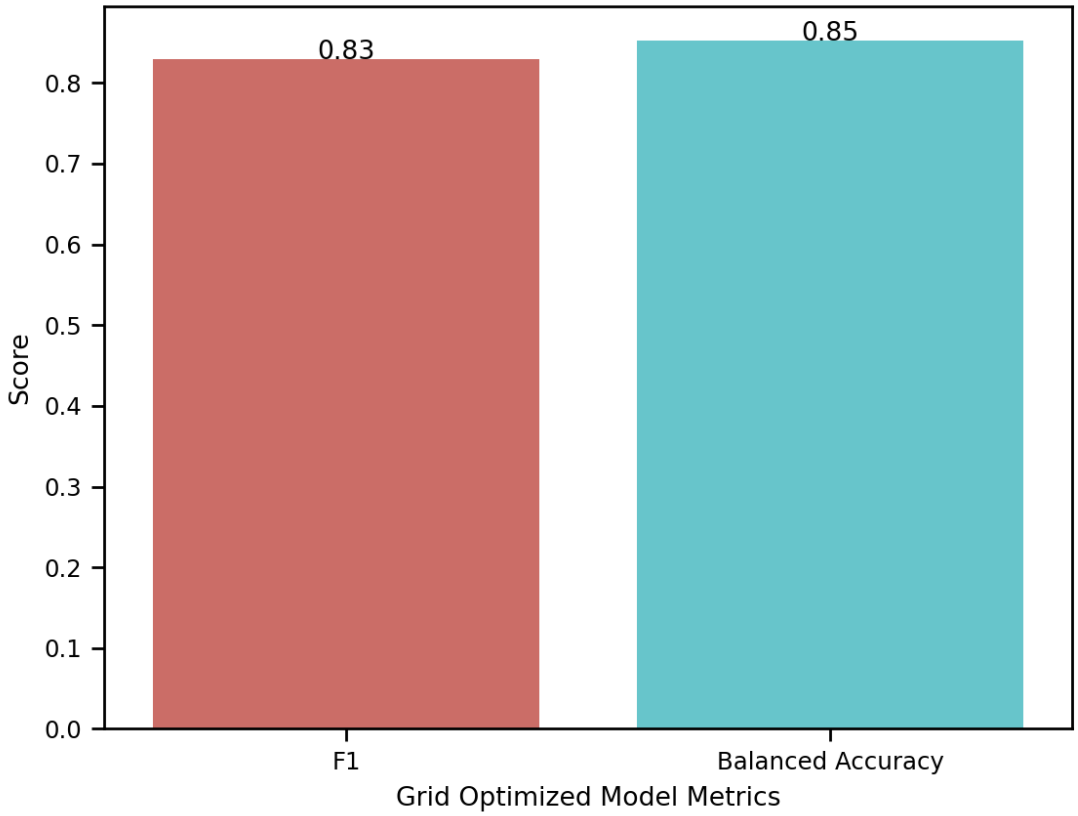

Figure 19: Grid Optimized Model results. Higher values represent better performing models.

5 Results Discussion

This chapter explores the interesting concepts and emergent ideas that stem from this work. It is divided into a discussion of the three core experiments performed, specifically, the regression, classification, and novel grid optimization experiments.

Regression Results Discussion

[23] achieved an RMSE of $\tilde{1}$75m in their optimized model. Their model used ML to predict an optimum scaling factor **S**. The linear regression model fit in this project is 100 meters less accurate than the optimized model used in [23]. The purpose of the test is not to achieve accurate predictions, but to identify if ML models can be viable for predicting bathymetry. Therefore, the accuracy of these models is less important than identifying the viability of the models. The training data used is predicted bathymetry, but shows that fitting a model to true bathymetry will yield a similar result.

Classification Results Discussion

The Random Forest model excelled with a balanced accuracy of 82%. Breaking down the results by class, the classifier predicted some classes with greater precision than others. This indicates that the decision boundary responded to certain trends in the data. In general, the Random Forest classifier performed better overall, which is why it was the best performing classifier for 47.2% of the world. This percentage can be seen in Figure 18.

The Bagging classifier performed on par with the Random Forest Classifier with a balanced accuracy of 79%. However, Figure 17 shows that the bagging classifier performed best around coastlines. This suggests that the model responds well to shallow waters. Shoreline data will also be consistently more accurate because of the proximity to land. Most of the world's high-resolution bathymetry is shoreline data, and it is possible that the model responded to the quality data.

The classification results show that labeling bathymetry can improve the performance of the models. In this work, it was tested that a random forest classifier can predict 82% of the world's predicted bathymetry within 150 meters of accuracy. However, the more interesting topic to analyze is the behavior of the models. The data used in training comes from aggregated external datasets and a predicted bathymetry dataset. The predictions of these models are being compared to predicted bathymetry, which represents the accuracy with relation to predicted values. This means that the accuracy in these models is not indicative of truly predicting global bathymetry. It does show that an ML model can be fit to data and be used to predict bathymetry, and if actually measured bathymetry and ocean features were used in training, the results would be comparable. Furthermore, parameter tuning, model selection, and dynamic feature selection could be used the increase the accuracy beyond current results.

Interestingly, some models excel along fault lines. For example, the decision tree classifier in Figure 17 performs well along what appear to be fault lines. However, when making predictions in the global scope the classifier is very inaccurate. This suggests that a classifier can be optimized using geospatial position.

Grid-Optimized Model Discussion

Grid optimization improved the accuracy of predictions by ~5%. These results are displayed in Figure 20. Obviously, the model selection and subsequent injection improved the results, creating an ensemble of many models and selecting them on demand. This could be caused by geophysical characteristics that benefit one classifier. For example, in Figure 17, the Decision Tree classifier performed best along what appears to be fault lines. It is possible that the characteristics related to being in proximity to a fault line benefited the model's decision boundary.

Clearly, Figure 17 provides evidence that model decision boundaries are sensitive to the features based on location. This leads to the theory that there is not a single best fit

model for predicting global bathymetry. Analysis of Figure 17 shows interesting consistencies that raise questions about the underlying features. For example, in Figure 17, the Bagging classifier appears to perform best around coast lines. This is consistent for most of the globe. However, in some coastal areas the Random Forest Classifier performs best. It is possible these areas are linked to port cities and the high-resolution bathymetry collected from shipping lanes.

Another interesting consistency is around fault lines. The coverages where the decision tree classifier performed best seems to follow fault lines. It is possible there is a geospatial attribute that contributes to this success, or a specific feature in these locations that contributes to the decision tree classifier's performance. The AdaBoostClassifier also performs well around fault lines, but to a lesser extent. It is possible there are trends in the data that explain why the AdaBoostClassifier shows this behavior.

Figure 20 represents the average prediction accuracy across all coverages where a model was the best-fit , which means that the model had the highest accuracy for predictions in that coverage. It is important to note that this is representing models that were trained on a reduce set of data, therefore, it is expected for the accuracy to be lower. What this graph shows is that the Random Forest Classifier had an average prediction rate of 51% across coverages. This is fairly consistent with what was expected as it was the best performing global model. However, other models that performed very poor on a global scale made accurate predictions in specific areas. For example, the Quadratic Discriminant Analysis classifier performed exceptionally well in specific areas, as well as the Ada Boost Classifier and the Artificial Neural Network classifier. With this being noted, selecting a optimal grid for prediction will allow the strongest classifier to be used at all times.

The evidence shows that an ML model for predicting bathymetry could be chosen based on geospatial location. Future work may include investigating the appropriate feature sets, coverage boundaries, depth boundaries, parameters, and metrics based on geophysical location. For example, volcanic activity creates new land. This activity has a causal relation-

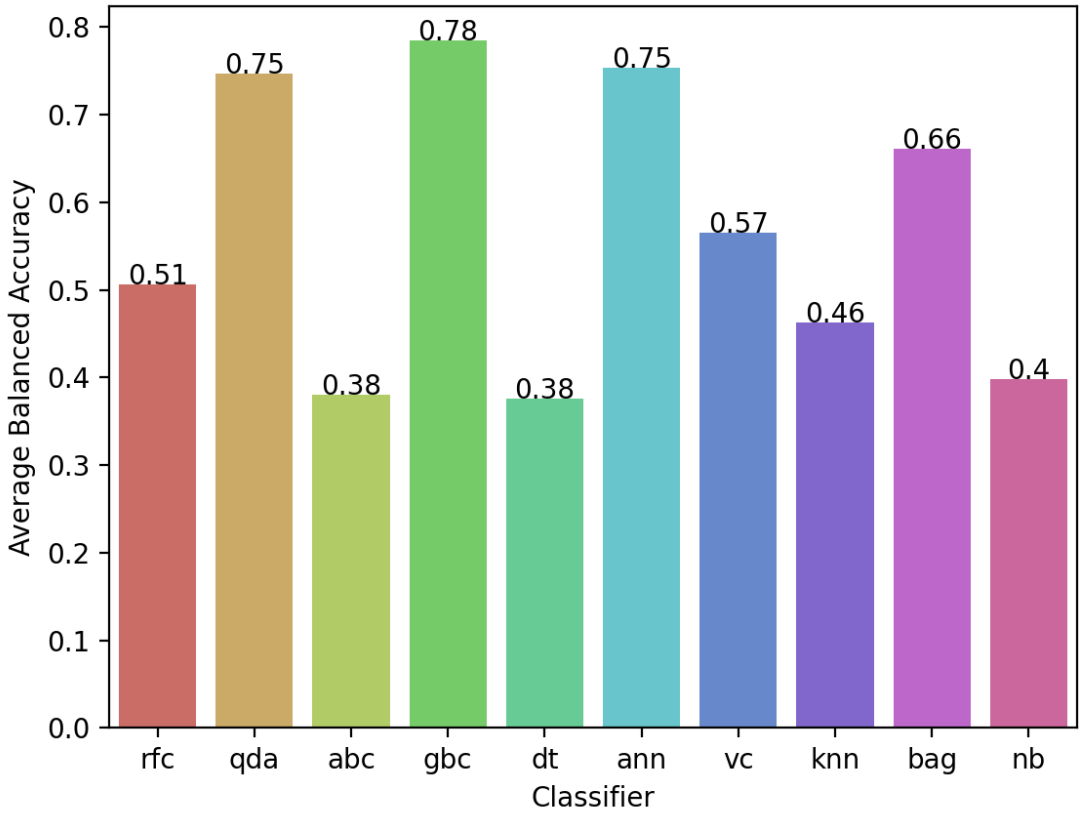

Figure 20: Average prediction accuracy of coverages where a model performed well. GBC had the highest average accuracy, but had a very small coverage.

ship to bathymetry, but volcanic activity at a specific point may not affect the bathymetry at a potential antipodal point. This is another example that the coverage boundaries are potentially a naive selection choice. It is possible that choosing models by performance across depth boundaries proves to be a better selector.

In this research, this optimization allowed each model to perform to its optimum. Each coverage highlighted distinct characteristics that could produce a better decision boundary. These coverages are simple partitions of the world, but could be extended in future work to optimize the selection.

General Discussion

The predictions made in this work are based on predicted bathymetry. Even with an 85% prediction rate, these models are not fit to compare with Earth Gravitational Model (EGM). The experiments in this work do show that there are accuracy gains to be achieved with model selection. Figure 17 gives evidence that there is not a best fit model for predicting bathymetry globally. Figure 20 gives evidence that an ensemble of some kind, optimizing by a decision function, will likely yield better results.

The grid optimization for model selection performed in this work is a novel approach for predicting bathymetry and provides evidence for optimal model selection with regards to predicting global bathymetry. It is reasonable to assume that selecting a model with a decision function will produce better predictions. Identifying the optimal decision function will help to explain why different models perform better or worse over the coverages. Grid optimization shows that a simple geospatial decision function can be used to improve theoretical prediction accuracy.

6 Conclusion

Machine Learning is a hot topic in academia. This paper explored the potential application of ML for predicting global bathymetry, as well as determining if there was evidence for a best-fit model. Many models were fit and compared leading to the determination that ML can potentially create viable models. However, limited access to accurate bathymetry and feature data temper expectations. This paper found there to not be a best fit model for predicting global bathymetry. Figure 17 shows evidence that models are more accurate dependent on location. In conclusion, while there is strong evidence that there is not a best fit global modal, ML can help determine an appropriate model based on an arbitrary decision boundary.

The code used to run most of the experiments in this work can be found on my Github at `https://github.com/nichipedia/masters-thesis-code`. All data used for learning can be downloaded from the sources listed in Table 1.

7 Future Work

There are many interesting avenues to explore from this research. A first avenue will be to perform the experiments at a higher spatial resolution. This work used two-minute spatial grids for representing all features and bathymetry. A two-minute grid was used for the memory and size advantages, allowing the computations and models to be run on a general-purpose work-station. Modern datasets for bathymetry are often represented in higher resolution grids. Naturally, the results will scale to a higher resolution.

Another interesting avenue will be to explore selecting features based on geophysical location. This could be tested by performing feature selection on the models across different coverages of the globe. A simple genetic algorithm could be run for each model. The resulting features will then be used for training and the best performing model will be selected with the optimum features. This experiment will eliminate features that do not locally support the model while highlighting the locally important features.

The same approach to feature selection could be taken to parameter tuning. Running a similar genetic algorithm to tune optimum parameters could potentially yield better results. Searching for optimal coverages is also has great potential.

Defining the best decision function for model selection is also a worthy investment. The core experiment in this project used a naive spatial boundary for selecting models. However, it is possible that geographic features were primary contributors to model success. Identifying several model selection functions and finding an optimal function will allow for conclusions to be drawn in regards to model success.

Finally, this work was performed using predicted bathymetry from the ETOPO dataset. As already stated, this was done in order to prove the viability of the models, not to prove the accuracy could be greater than an EGM. However, this is an experiment that can be executed. Training these models against true bathymetry and comparing the metrics to existing EGMs will give an indication of the ability to predict bathymetry.

References

[1] Harold K Farr. "Multibeam bathymetric sonar: Sea beam and hydro chart". In: *Marine Geodesy* 4.2 (1980), pp. 77–93.

[2] Timothy H Dixon et al. "Bathymetric prediction from Seasat altimeter data". In: *Journal of Geophysical Research: Oceans* 88.C3 (1983), pp. 1563–1571.

[3] National Geophysical Data Center. *ETOPO-5 bathymetry/topography data*. 1988.

[4] Naomi S Altman. "An introduction to kernel and nearest-neighbor nonparametric regression". In: *The American Statistician* 46.3 (1992), pp. 175–185.

[5] Stuart Geman, Elie Bienenstock, and René Doursat. "Neural networks and the bias/variance dilemma". In: *Neural computation* 4.1 (1992), pp. 1–58.

[6] Walter HF Smith and David T Sandwell. "Bathymebibtric prediction from dense satellite altimetry and sparse shipboard bathymetry". In: *Journal of Geophysical Research: Solid Earth* 99.B11 (1994), pp. 21803–21824.

[7] Yoav Freund and Robert E Schapire. "A desicion-theoretic generalization of on-line learning and an application to boosting". In: *European conference on computational learning theory*. Springer. 1995, pp. 23–37.

[8] Donald Ervin Knuth. *The art of computer programming*. Vol. 3. Pearson Education, 1997.

[9] Walter HF Smith and David T Sandwell. "Global sea floor topography from satellite altimetry and ship depth soundings". In: *Science* 277.5334 (1997), pp. 1956–1962.

[10] Jihoon Yang and Vasant Honavar. "Feature subset selection using a genetic algorithm". In: *Feature extraction, construction and selection*. Springer, 1998, pp. 117–136.

[11] Leo Breiman. "Random forests". In: *Machine learning* 45.1 (2001), pp. 5–32.

[12] Harry Zhang. "The optimality of naive Bayes". In: *AA* 1.2 (2004), p. 3.

[13] Payal Parekh, Michael J Follows, and Edward A Boyle. "Decoupling of iron and phosphate in the global ocean". In: *Global Biogeochemical Cycles* 19.2 (2005).

[14] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

[15] Hendrik L Tolman. "The 2007 release of WAVEWATCH III". In: *Proc. 10th Int. Workshop of Wave Hindcasting and Forecasting*. 2007.

[16] Nikolaos Pavlis et al. "Earth gravitational model 2008". In: *SEG Technical Program Expanded Abstracts 2008*. Society of Exploration Geophysicists, 2008, pp. 761–763.

[17] JJ Becker et al. "Global bathymetry and elevation data at 30 arc seconds resolution: SRTM30_PLUS". In: *Marine Geodesy* 32.4 (2009), pp. 355–371.

[18] Eric P Chassignet et al. "US GODAE: global ocean prediction with the HYbrid Coordinate Ocean Model (HYCOM)". In: *Oceanography* 22.2 (2009), pp. 64–75.

[19] Gavin C Cawley and Nicola LC Talbot. "On over-fitting in model selection and subsequent selection bias in performance evaluation". In: *Journal of Machine Learning Research* 11.Jul (2010), pp. 2079–2107.

[20] Ryan N Smith et al. "Planning and implementing trajectories for autonomous underwater vehicles to track evolving ocean processes based on predictions from a regional ocean model". In: *The International Journal of Robotics Research* 29.12 (2010), pp. 1475–1497.

[21] Chih-Lin Wei et al. "Global patterns and predictions of seafloor biomass using random forests". In: *PloS one* 5.12 (2010).

[22] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[23] Babula Jena et al. "Prediction of bathymetry from satellite altimeter based gravity in the Arabian Sea: Mapping of two unnamed deep seamounts". In: *International Journal of Applied Earth Observation and Geoinformation* 16 (2012), pp. 1–4.

[24] Lars Buitinck et al. “API design for machine learning software: experiences from the scikit-learn project”. In: *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*. 2013, pp. 108–122.

[25] Gabi Laske et al. “Update on CRUST1. 0—A 1-degree global model of Earth’s crust”. In: *Geophys. Res. Abstr.* Vol. 15. EGU General Assembly Vienna, Austria. 2013, p. 2658.

[26] D Sakellariou et al. “Preliminary report on underwater survey in the Farasan Islands by the R/V Aegaeo, May–June 2013”. In: (2015).

[27] Poliyapram Vinayaraj, Venkatesh Raghavan, and Shinji Masumoto. “Satellite-derived bathymetry using adaptive geographically weighted regression model”. In: *Marine Geodesy* 39.6 (2016), pp. 458–478.

[28] Leo Breiman. *Classification and regression trees*. Routledge, 2017.

[29] Andrzej Chybicki. “Three-dimensional geographically weighted inverse regression (3GWR) model for satellite derived bathymetry using Sentinel-2 observations”. In: *Marine Geodesy* 41.1 (2018), pp. 1–23.

[30] Thomas Meissner, Frank J Wentz, and David M Le Vine. “The salinity retrieval algorithms for the NASA Aquarius version 5 and SMAP version 3 releases”. In: *Remote Sensing* 10.7 (2018), p. 1121.

[31] ID Scope. “Single and Multibeam echo-sounders”. In: (). URL: `http://www.id-scope.mc/Geophy03_EN.html`.

Vita

The author was born in Covington, Louisiana. He acquired his bachelors degree in Computer Science from Southeastern Louisiana University in 2017. He is a Computer Scientist for the Naval Research Laboratory who specializes in Machine Learning, Web Scraping, and Software Engineering.