Contents lists available at ScienceDirect

Deep-Sea Research Part I

journal homepage: www.elsevier.com/locate/dsri

Machine learning model selection for predicting bathymetry

Nicholas Moran[a,b], Ben Stringer[b], Bruce Lin[b], Md Tamjidul Hoque, Ph.D.[a,*]

[a] *Department of Computer Science, University of New Orleans, New Orleans, LA, 70148, USA*
[b] *US Naval Research Laboratory, Stennis Space Center, MS, 39529, USA*

ARTICLE INFO

Keywords:
Machine learning
Predicting bathymetry
Earth gravitational models
Classification
Genetic algorithms

ABSTRACT

This research investigates the viability of using Machine Learning (ML) for predicting bathymetry. We built and trained several models using ocean features aggregated from multiple sources and predicted bathymetry from the ETOPO dataset at a 2-min resolution. Each model was evaluated to identify a global best fit, however we found that none performed well on a global scale. When training on subsets of the world, we observed that some models performed significantly better, which led to developing a novel model selection technique that identifies the best performing model and most relevant features for a given geospatial coverage. This leads to improved predictions and more reliable results. This model selection technique can be generalized to be applied to any set of models.

1. Introduction

The forefront in global bathymetry mapping is the aggregation of measured and predicted sources. Measured bathymetry comes from echo-sounders, which generate an accurate and high-resolution bathymetry grid. However, surveys using echo-sounders are expensive, resulting in very limited global coverage of accurate measurements. Specifically, only 10% of the world has been mapped with echo-sounders (Becker et al., 2009). The remainder of the oceans' bathymetry must be predicted. These come from the earth Gravitational Model (EGM), the standard for estimating global bathymetry (Becker et al., 2009; Smith and Sandwell, 1994, 1997; Smith et al., 2010).

Predicting bathymetry is a complicated problem that involves several unknowns. Very little accurate data is available for training due to the vast nature of the earth's oceans. The majority of ocean feature data is either interpolated or predicted. Modeling bathymetry as a function of gravity is an excellent approach at coarse resolutions since it relies on the results of global measurements rather than interpolated data, however, prediction error becomes an issue at finer resolutions. Modern EGM predictions have an error of approximately 180 m (Jena et al., 2012).

Our research focuses on identifying whether there is a global best fit model and investigates ways to optimize theoretical predictions. We implemented a genetic algorithm, which relies upon the principles of evolution (Yang and Honavar, 1998), to select the most relevant features. We chose this approach because it would have been impractical to examine every feature combination exhaustively, and genetic algorithms allow for near-optimal feature selection in a reasonable time frame. The known data used in this research 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 m (Becker et al., 2009). Therefore, the data from EGMs is only sufficient for creating theoretical models.

We were not able to identify a model that performed well globally. This led to the development of a novel model selection approach we call the Grid Optimized Ensemble. This approach gives us modest improvements and lays the groundwork for a number of future optimizations.

2. Background

This section provides the necessary background for this paper. We discuss bathymetry data and the current state of the art for bathymetry prediction models, as well as machine learning concepts, including the various types of learning, how machine learning could be applied to bathymetry prediction, and how to evaluate the results.

2.1. Bathymetry

Much of the world's oceans are not mapped with high accuracy surveys (Becker et al., 2009). Approximately 90% of the oceans have not been surveyed, making the depth of the earth's oceans a perplexing

* Corresponding author.
E-mail addresses: npmoran@uno.edu, npmoran@uno.edu (N. Moran), ben.stringer@nrlssc.navy.mil (B. Stringer), bruce.lin@nrlssc.navy.mil (B. Lin), thoque@uno.edu (M.T. Hoque).

https://doi.org/10.1016/j.dsr.2022.103788
Received 31 August 2021; Received in revised form 15 April 2022; Accepted 21 April 2022
Available online 2 May 2022

mystery. In general, bathymetry, that is, the depth of a body of water at a given point, can either be measured or predicted. There are several techniques for obtaining accurate bathymetric measurements. They can be classified by direct measurements or remote sensing techniques. Bathymetry collected by Echo Sounders is an example of direct measured Bathymetry, Sattelite Derived Bathymetry, and Sattelite Altimetry Earth Gravitational Models are examples of remotely sensed bathymetry.

Echo-sounders operate by monitoring sound as it passes through a column of water. The relationship between the height of the water column and the time required for sound to return will be directly related to the depth. This method is very accurate, resulting in less than a meter of error. Single-Beam echo sounders use a concentrated beam of sound to gain a high-resolution image of the seafloor. While this is ideal for detecting objects, the width of the recorded bathymetry is relatively small, at only a few meters in width. Multi-beam echo sounders, on the other hand, can achieve a track width of up to 2 km. They do not provide a high-resolution image of the seafloor, but they do record accurate bathymetry and are preferred for survey missions because of their wide coverage.

Satellite Derived Bathymetry is a remote sensing approach that uses water's attenuation to measure depth, however this approach only works in shallow water precisely because of that attenuation. There are techniques that improve on this to achieve higher accuracy in shallow waters with SDB. Two Media Photogrammetry corrects for the effects of refraction by observing the same bathymetry point from two sources (Cao et al., 2019). This method can be further improved by utilizing Machine Learning (Agrafiotis et al., 2020).

The standard for predicting bathymetry was introduced by Smith and Sandwell, 1994, 1997, which uses Earth Gravitational Models (EGMs) for predicting bathymetry. Their approach predicts depth using the correlation between sea surface altimetry and geoid height. Sea surface altimetry is measured by satellites. The height of a sea swell is then mapped to an inferred gravity of an underwater geoid. This method is generally accepted as best practice for predicting globally, but it can often have a large error, resulting in inaccurate data. There are a number of variations on this model, each with its own strengths. The primary difference between them is the resolution at which the bathymetry is predicted.

An issue with EGMs is that bathymetry does not correlate directly with sea surface altimetry. There are many factors that introduce errors to that correlation. To overcome these factors, a scaling factor is used in the prediction function. Machine Learning has been shown to improve the error of these prediction models by optimizing that scaling factor. For example, Jenna et al. (Jena et al., 2012) used machine learning to identify a scaling factor for optimizing bathymetry prediction models. Specifically, they trained models to predict an optimal scaling factor that was used to correct for differences in regional sediments and geoid properties. Smith and Sandwell (1994) introduced scaling factors but calculated them programmatically. Jenna et al. (Jena et al., 2012) introduced machine learning for optimizing this scaling factor which showed to improve its effectiveness. More importantly, they surveyed areas of the Arabian Sea and calculated an average error of 180 m with known predicted bathymetry in those areas.

2.2. Machine learning

Machine Learning (ML) is the use of trained models to predict a value. ML approaches can be cataloged into several different categories. They may be supervised or unsupervised depending on whether the training data is labeled with the values the model should predict. Supervised ML models can be further subdivided into regression or classification. Regression models attempt to predict a continuous value, while classification models attempt to predict a label from a set of discrete values.

Regression would seem to be the natural approach for predicting bathymetry. After all, the depth of the ocean at any point is a continuous value. Our training data, however, is itself a predicted value, and as noted in (Jena et al., 2012), it is subject to an average error of 180 m. Therefore, as we discuss in detail in Section 3, we modeled this problem as a classification problem.

Some of the classification models are binary, that is, they consider two labels: positive and negative. In this case, multi-class classification operates by iterating over the range of labels, setting the current label as the "positive" case and all others as the negative case.

Classification over a continuous range is achieved using an approach called "binning." The range of possible values is divided into bins, not necessarily of fixed size. The training labels are converted to represent their respective bin, so a training value of 1900 m could be given the label (2000–1850 m).

In addition to labels, an ML model needs data related to the label to aid in prediction, called features. Often, there are a number of features to choose from, and it is not always clear which will provide the most value. Some features will be noise and unnecessary for the final prediction, which will slow down the model's execution. Identifying the appropriate features is known as feature selection.

There are a number of valid approaches for approaching feature selection, including grid search, dimensional analysis, simple variable correlations, and genetic algorithms. Grid search is an exhaustive search through the feature space, attempting all combinations of features to determine which provide the most benefit. Dimensional analysis and simple variable correlations examine relationships between features with the goal of reducing the feature space. Due to the number of features we examined in this work, these approaches either took too long or simply did not offer enough improvement to the model.

This work used a genetic algorithm approach for feature selection (Yang and Honavar, 1998). The genetic algorithm approach gave relatively quick model improvements with little effort. Fig. 3 shows the basic flow of a general genetic algorithm. This can be expressed in terms of states. At the initial state, a population is created. In this work, the population is some combination of possible feature selections. After initialization, each member's fitness is assessed, that is, each permutation of features are evaluated by using them to execute a model. Next, members are sorted by their fitness score, and some number of top-scoring members are carried forward. It is not required that only high-performing members are carried forward, however since the goal is to identify the optimal feature set, we don't want to omit the high performers. Next, the population is replenished by combining carryover members to generate new population members. A random mutation follows this to a subset of the population. This process repeats until some termination condition is met, where the top performer is selected as the optimal set. It is important to note that the genetic algorithm is not deterministic. Due to the randomness, two runs could result in very different results. However, when the search space is too large for an exhaustive search, the genetic algorithm makes a good compromise.

Validating Machine Learning models helps ensure that a model does not overfit training data. Overfitting occurs when a model is only able to predict data that it has encountered during training. Validation addresses this by evaluating a model's effectiveness at predicting data it has not seen before. In order for this to work, some of the labeled training data must be held out of the training process so that the model does not have the opportunity to learn from it.

k-Fold cross-validation is an effective way for validating models. It works by dividing a training set into k groups, called "folds." k iterations are then executed. At the i-th iteration, the i-th fold is withheld. The model is trained on folds $j \in [1, k]$, $j \neq i$ and then evaluated using i-th fold. The testing results are then averaged across the k iterations, which results in a validation score. A common value for k is 10, which would be referred to as 10-fold cross-validation.

An ML pipeline defines the steps and processes used for producing a prediction model end to end. A typical pipeline would include loading various data sources, cleaning the data by assigning default values or removing outliers, normalizing the data to some standard range, typically [0,1], training a model, and evaluating the results. The benefit of defining a pipeline in this fashion rather than only operating on clean data is that the entire process is repeatable. Perhaps the amount of data is too large for processing on commercial hardware, but working with a subset can provide the researcher with confidence that the algorithm is working. The completed pipeline can then be transferred to equipment suitable for the large data without requiring additional work. In addition, more data could become available well after the initial pipeline was created. If the pre-processing steps were not included in the pipeline, it may not be clear how the new data should be processed to make it compatible with the existing models.

Scikit Learn is an open-source library developed by the Python community (Pedregosa et al., 2011). It exposes an intuitive Application Programming Interface (API) and a 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, we implemented a genetic algorithm component for feature selection using the sklearn API. This component was then able to be used seamlessly with all existing models and other sklearn components.

2.3. Metrics

In order to evaluate a model and determine its usefulness, we must calculate some metrics. There are a variety of useful ML evaluation metrics. For this research, we relied primarily on R^2, F1-score, and Balanced Accuracy. All metrics, as shown, are for a binary classification problem. For use in a multi-class classification problem, each metric is applied to a class, and then all calculations are averaged.

The R^2 metric, also known as the coefficient of determination, is a measure of variance in the dependent variable that is predictable from the independent variables. The R^2 score reflects how useful the features of a model are as predictors, with higher values indicating higher utility. For a given training dataset with n entries, where an entry represents the set of features and the label for a given geospatial location, we can use Equation (4) to calculate the R^2 metric. The value y_i represents the known labels, f represents the predicted label, and μ represents the mean of the known labels.

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

For a classification model, training data is labeled into classes and the model predicts a class label for new input data. Predictions that correctly define a class are labeled as True, and predictions that incorrectly define a class are labeled as False. To simplify the following metrics, the equations will be defined in terms of binary classification where there are two classes, Positive and Negative. Precision, known as the true positive rate, is shown in Equation (5). It is a measurement of how good a model is at avoiding false positives. Recall, shown in Equation (6), is a measurement of how good a model is at avoiding false negatives. The F1-Score, shown in Equation (7), represents a harmonic mean of a model's precision and recall.

$$precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \tag{5}$$

$$recall = \frac{True\ Positives}{True\ Positives + False\ Negatives} \tag{6}$$

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

The Balanced Accuracy matric is useful for datasets with imbalanced labels, that is, a given label has a significantly larger or smaller number of examples than another. For example, consider a classification problem with 100 samples, 95 of which are positive, and 5 are negative. A classifier could predict all 100 as positive, giving it an impressive 95% accuracy, despite performing so poorly against the negative cases. Balanced accuracy, shown in Equation (8), will give a score of 50%, which is a better indication of the success of that classifier's results.

$$Balanced\ Accuracy = \frac{True\ Positive\ Rate + True\ Negative\ Rate}{2} \tag{8}$$

The Pearson Correlation Coefficient (PCC) is the covariance of two variables divided by the product of their standard deviations. The value ranges between −1 and 1 and implies a linear relationship between the two variables. A value closer to 1 implies a direct relationship between the variables. A value closer to −1 will denote an indirect relationship between the variables. A value of 0 will imply no linear relationship between the variables.

3. Methodology

This section details the approach taken in this research. In Section 2, we described a general machine learning pipeline. This section describes the various stages of the pipeline used in this research, which can be divided into two main phases, the data pre-processing phase, and the learning phase.

The data pre-processing phase involves all of the steps necessary to begin using the data. We begin by describing the data used in this effort, including the feature data as well as the bathymetry labels, which are the values we are trying to predict. We describe how the relevant ocean features were selected from a large set of aggregated features. This step involves aggregating and gridding the features, cleaning the data by interpolating missing values and masking land, and identifying the relevant features with a genetic algorithm.

The learning phase involves building models using the training data and validating their predictions using 10-fold cross-validation. We describe our use of the trained models to make predictions with the goal of identifying a global best fit. Finally, we describe the Grid Optimized Ensemble, a novel model selection approach that gives modest performance improvements.

3.1. Data representation

All data used in this research was placed into grid structures. Naturally, the data represent geospatial locations on earth, and 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 grid representation provides a standard that allows multiple grids with different sets of data to reference the same geospatial areas.

The spatial resolution of a grid defines its coverage and can be described by a grid cell's height and width. The cell 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. Despite this, the grid format is preferred because it allows data from multiple disparate sources to be aggregated consistently and structured.

All data in this research has been organized into 2-min bathymetry grids. A 2-min bathymetry grid has a spatial resolution of 0.034° per cell, which is approximately 3 km of spatial coverage. The grids have a column count of 5400 cells and a row count of 10,800 cells. This resolution was chosen for experiments to conserve memory and time. Larger grids have an exponentially larger memory and computational footprint. We used the ETOPO2v2 (National Geophysical Data Center, 1988) dataset as the source of the 2-min bathymetry grid. Finer resolution datasets exist, such as the SRMT30 (Becker et al., 2009) at the 30-s resolution; however, the 2-min resolution offers a good balance of memory, accuracy, and computational costs.

3.2. Bathymetry interval labels

The ETOPO 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 for this research because of the 2-min resolution it offers. ETOPO was aggregated by the National Geophysical Data Center (NGDC), a National Oceanic and Atmospheric Administration department.

Land topography is included in the ETOPO dataset. This proved to be an issue for creating accurate bathymetry predictions; therefore, a mask was created to remove the land topography. This is applied to all data before training to ensure that only ocean data is used in training. The mask was constructed of Boolean values to mark cells for inclusion or exclusion. If a cell contains a majority of land, it will be marked for exclusion. This approach worked well for masking land for training.

Ocean features were aggregated from several studies and placed into 2-min bathymetry grids. This data's gridding was done so that data in a specific geospatial area could easily be referenced to data from a different source in the same spatial area.

The continuous bathymetry data from the ETOPO dataset was used for training data. To facilitate this for classification, the ETOPO dataset was binned into discrete classes. This binning was performed at 150-m intervals. This effectively gives a bathymetry label an error of 150 m. This error was desired so that it could be compared to the results from Jenna et al. (Jena et al., 2012).

3.3. Feature data

The feature data used in this research was aggregated from the various ocean and earth studies. This data was normalized and converted to the grid format so that it could be used together. Missing data points were either interpolated or given default values. Our goal was to gather a large set of features regardless of the data's immediate relevance. In general, the features examined in this research were considered for their potential as predictors.

The normal approach for predicting physical planetary phenomena is to develop a mathematical model that focus on the underlying physical processes involved in the phenoma. For ocean bathymetry, seamounts are a factor in the depth of our oceans and their height can be mathematically modeled by sea surface altimetry which correlates to the gravitational pull of the seamount. That correlation is the backbone of the mathematical modeling approach known as EGMs. We approached this from a data science perspective. That is, rather than attempting to define a model relating some feature, e.g., fish biomass, with bathymetry, we used the tools provided by ML to identify which features were related. From this, we can examine the relationship further to determine if the correlation implies causation and whether a more refined model could be built.

Our synthetic datasets offered an effective tool for experimenting with ML in this problem space. Ideally, if some particular feature data introduces significant noise, then it will be removed during feature selection. Our initial work began with 79 potential features. The features deemed most relevant, along with their originating study, are shown in Table 1.

A detailed analysis of the relationship between these features and their utility in predicting bathymetry is out of the scope of our work, however, we will note that the features selected by the genetic algorithm are those with stronger Pearson correlation coefficients.

Fig. 1 shows a plot of estimated fish biomass against bathymetry. This positive linear relationship is easy to see in the graph and is reflected in the Pearson Correlation Coefficient (PCC) value of 0.6834. 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.

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

Feature	Origin Study
Mantle Density	CRUST1 (Laske et al., 2013)
LAND One Hot	ETOPO (National Geophysical Data Center, 1988)
Crust Thickness	CRUST1 (Laske et al., 2013)
Low, Mid, High Crust Density	CRUST1 (Laske et al., 2013)
Estimated Current East, North, Mag	HYCOM (Chassignet et al., 2009)
Sea Nitrate, Phosphate, Salinity Measurements	NASA Studies (Meissner Frank and Le Vine, 2018; Parekh et al., 2005)
Sea Temperature, Silicate Measurements	NASA Studies
Sediment Thickness	CRUST1 (Laske et al., 2013)
BioMass Features	CRUST1 (Wei et al., 2010)
Geoid Features	EGM (Pavlis et al., 2008)
Wave height, period	WAVEWATCH (Tolman, 2007)

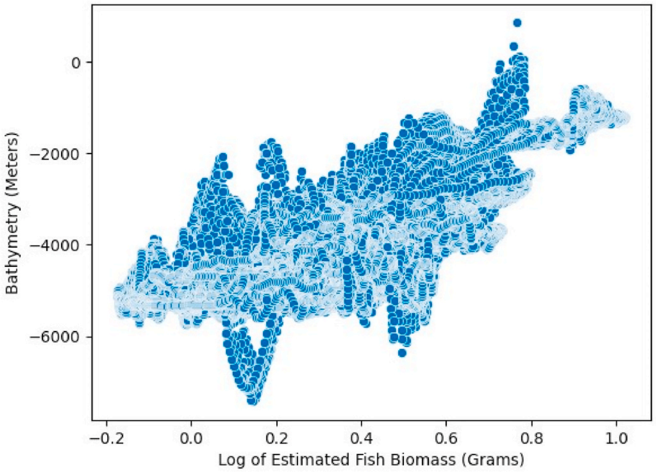

Fig. 1. Graph of Bathymetry and Estimated Fish BIOMASS. Bathymetry is measured in meters, and Fish Biomass is measured in grams. The Pearson correlation coefficient between these values is 0.6834, indicating a strong positive linear relationship.

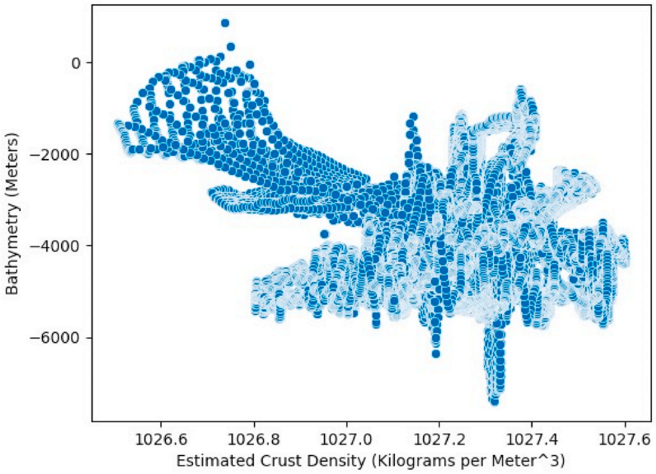

Fig. 2. Graph of Bathymetry and Estimated Crust Density. Bathymetry is measured in meters, and Crust Density is measured in milligrams per squared centimeter. The Person correlation coefficient between these values is −0.5425, indicating a strong negative linear relationship.

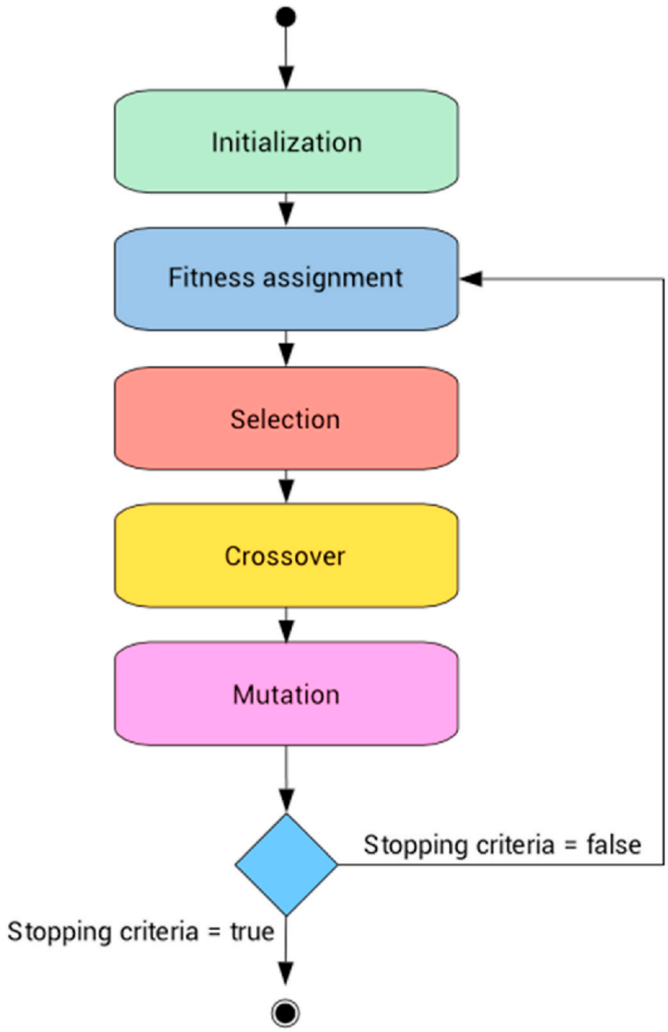

Fig. 3. State diagram of general genetic algorithm.

Fig. 2 shows a graph of crust density against bathymetry. A negative linear relationship can be seen in the graph, which is reflected in the PCC value of −0.5425. 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 cannot be used to describe the correlation of the variables.

The relationship between bathymetry and features such as estimated oxygen, nitrogen, and salinity may have intuitive explanations, while other features, such as crust density, may not be so easily explained. The challenge with the data science approach is in explaining why certain features were selected. While we don't address that problem here, future studies in the relationship between these features and their relationship to bathymetry will be necessary.

3.4. *Feature selection*

Our genetic algorithm for feature selection is a simple implementation. The population is represented by a set of binary strings, randomly initialized. Each string has a character length equal to the number of features in our feature space, i.e., each string is 79 characters long. Each character represents whether a feature is active or not. We then trained a Random Forest model using each feature combination from the population. The resulting model's accuracy using 10-fold cross-validation represents the fitness of that combination of features. Selection is performed by choosing the top 20% of models whose feature combinations performed best and passing those combinations onto the next generation. A simple crossover mutation of the strings is used to replenish the population along with a modest 5% mutation rate. The algorithm terminates when an accuracy of greater to or equal to 75% is reached. We selected the feature combination whose model performed best as the feature set for the remainder of this effort. The winning features are shown in Table 1.

3.5. *Learning methodology*

Regression is an intuitive approach to predicting bathymetry. We trained three regression models to test the effectiveness. However, we found that the regression models from SK Learn performed poorly. Therefore, the reminder of our effort focused on classification models.

Classification models were chosen over regression for two reasons. First, regression models output a continuous value. Intuitively this seems like the ideal approach, however, we note that we don't actually know the true bathymetry for much of the world's oceans. Therefore, trying to train a model to predict a value that itself has 180 m of error does not make as much sense. Our goal was not to predict true bathymetry since we aren't able to validate this. Our goal was to determine if a single model could be used to make predictions for the entire world. Predicting a bin with a range of 180 m was sufficient to meet this goal. Second, we found classification models were easier to work with and simply performed better than regression models.

To perform classification, models must be trained using discrete values. We mapped the ETOPO data from continuous values to discrete labels by creating binned ranges. This proved to be trivial due to the ordered nature of bathymetry. The range of each bin was the same, 150 m, and there were a total of 128 bins.

Models were evaluated using 10-fold cross-validation with Balanced Accuracy as the scoring function. 10-fold cross-validation was chosen because it offered a very quick validation compared to other approaches. The models trained in this project were imported from the Sklearn library and are listed in Table 2.

3.6. *Model selection methodology*

The original goal of this effort was to determine if a best-fit model for global predictions could be identified. This involved training models against global coverage and evaluating their performance. We trained 10 models and evaluated them using 10-fold cross-validation as discussed in the previous section, and used the validated performance metrics to compare models. The outcomes are covered in the Results section, however, in short, it was observed that many models performed poorly on a global scale.

This led to an investigation of whether locally optimum models could be used for predictions, that is, whether there are geospatial areas where a particular model outperforms other models. To perform this test, we split the world into 4050 geospatial coverages. We trained the full set of models on the data for those areas and validated the results. The model that was most successful in a particular area was then recorded. The results of this experiment lead to the creation of a novel model selection technique called "Grid Optimization." Instead of expecting a single model to perform well globally, the optimized grid approach chose the most appropriate model based on the region. This led to an improvement in global prediction performance.

Table 2
List of Classifier Models trained and evaluated for this project.

Random Forest (RFC)	Bagging (BAG)
Decision Tree (DT)	*k* Nearest Neighbors (KNN)
Multi-Layer Perceptron (MLP)	Ada Boosting (ABC)
Gaussian Naïve Bayes (GNB)	Voting Classifier (VC)
Support Vector Classifier (SVC)	Quadratic Discriminant (QDC)

4. Results

This section details the results of our experiments. We begin by looking at the global models and discussing the top performers. We then look at the locally trained models and identify which performed well in each coverage. Finally, we discuss the Grid Ensemble Global Model and compare its results against the standard global models.

4.1. Global models

We trained and evaluated 10 models using the features listed in Table 1. The models we trained are listed in Table 2. The Voting, SVC, Gradient Boosting, and QDA classifiers failed to complete training due to the amount of data involved. These were trained on reduced sets of data, which we detailed in Fig. 4. They were hand-chosen to include features from across the globe in different environments and hemispheres. It is important to note that we could not get the Support Vector Classifier (SVC) to train with a reduced dataset.

We used F1 and balanced accuracy metrics to evaluate their performance, which are shown in Fig. 5. As we can see, the Random Forest and Bagging classifiers were the highest performing models. KNN also performed well, however, this result is suspect since we were unable to fully train it. With a balanced accuracy of 0.47, the Decision Tree classifier performs so poorly that it should not be used for predictions. The remaining classifiers performed even worse.

4.2. Local models

In order to get a better understanding of these models' performances, we divided the world into coverages. The size and shape of the coverages were chosen arbitrarily, large enough to enable quick calculations but small enough to visualize trends. We then trained and evaluated each of the ten models against each coverage. We determined which performed best in the region based on the F1 score and the balanced accuracy metric. The results of this experiment are shown in Fig. 6. Each square represents a coverage, which is color-coded based on the best performing model for that coverage. Fig. 7 shows the percent of coverages where a particular model performed best. As expected, the Random Forest and Bagging classifiers dominated, however roughly a third of the world was best predicted by one of the other models. These models, when trained globally, performed too poorly to be of any use, however, they performed well in very specific instances.

Examining Fig. 6 more closely, we notice that although the Random Forest classifier performed best overall, the bagging classifier performs well around the shallow coastlines, and the Decision Tree classifier performs well along fault lines. This suggests that each model's decision boundary is responding to certain trends in the data.

4.3. Regression results

We trained three regression models to predict bathymetry. Table 3 shows the metrics for the regression models. The R Squared score shows a strong correlation with the underlying data. However, the error of the models is large. This large error made using regression impractical for our experiments. An accurate prediction from the classification models has an error of 150 m. This offered far better accuracy compared to the regression error shown in Table 3.

4.4. Grid Ensemble Global Model

We developed an ensemble method that selected the most appropriate model for coverage based on the results shown in Fig. 6. This approach resulted in modest improvements to the global prediction results, shown in Fig. 5. As this figure shows, the Random Forest classifier, the best performing global classifier, had an F1 score of 0.81 and a balanced accuracy score of 0.82 for global predictions, however, the grid optimized ensemble method brought that value up to 0.83 and 0.85, respectively.

Fig. 8 shows the average balanced accuracy score for the regionally trained models. In this figure, the results are only included when that model is selected as the best fit for that region. So, for example, although the Gradient Boosting had the highest average balanced accuracy, Fig. 8 shows that it was selected for less than 1% of the world.

The code and data related to this research can be found here, https://github.com/nichipedia/masters-thesis-code.

5. Conclusions

In this work, we detailed our attempt to identify a machine learning model capable of predicting global bathymetry. We trained a collection of models and evaluated the results using the ETOPOv2 dataset, which is itself predicted bathymetry. We were not able to identify a single model that is a best fit for predicting global bathymetry, however, this work demonstrates that while some models perform poorly for global predictions, they still perform well when used in specific areas. We concluded that some feature and model combinations perform well together, which led to the creation of an optimal model selection technique, the Grid Optimized Ensemble. When applied, this technique leads

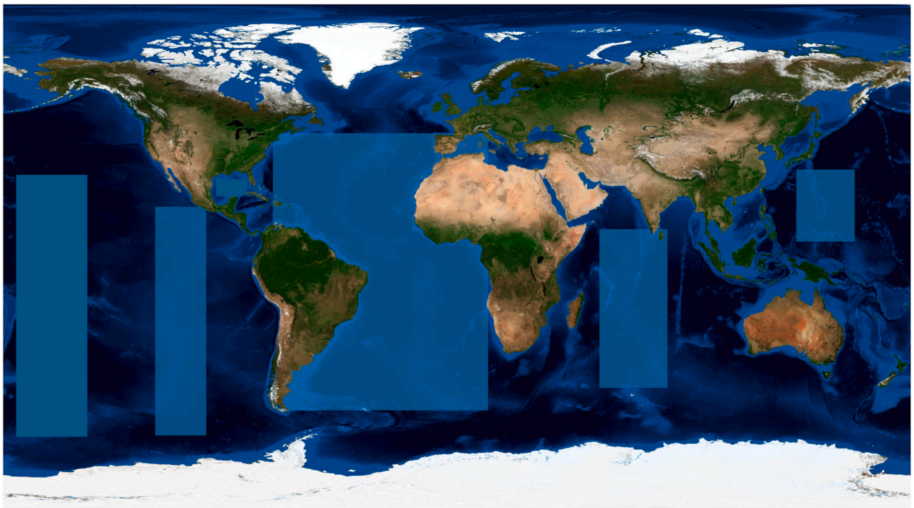

Fig. 4. List of reduced datasets.

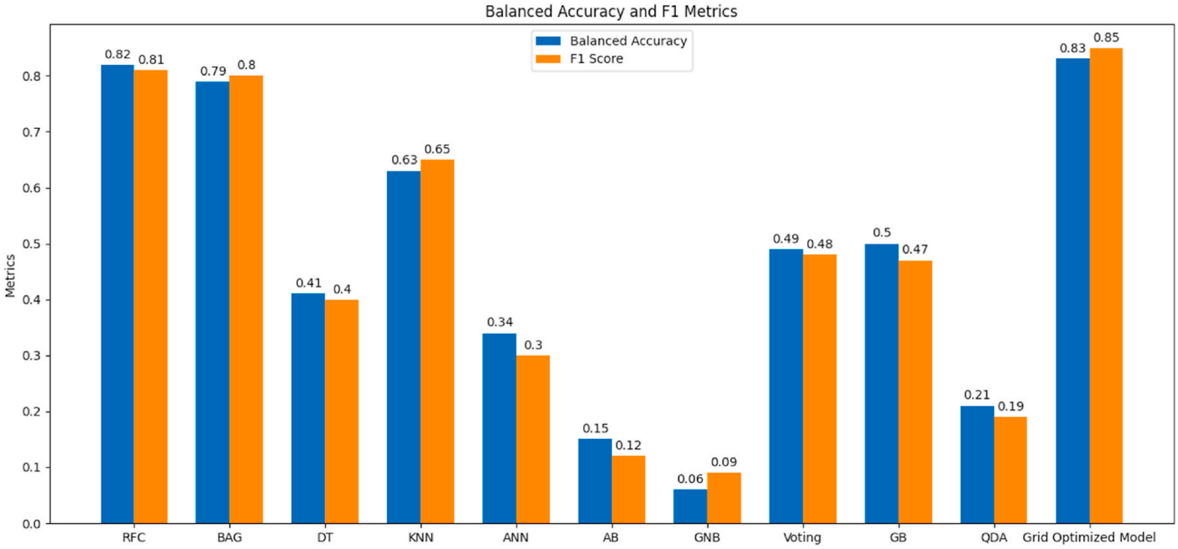

Fig. 5. Balanced Accuracy and F1-Score metrics for all models.

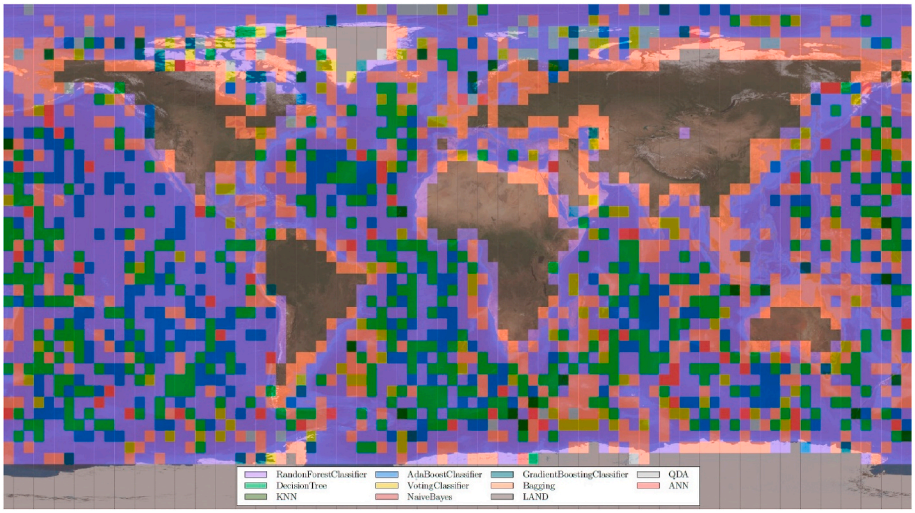

Fig. 6. World Coverages and Successful Models. Each square represents a coverage. The shaded color represents the model that was most successful in that coverage.

to a modest increase in accuracy over models trained on global features. Finally, we acknowledge that most of this work has been exploratory, lacking the rigor needed to make definitive statements about the usability of a particular model or combination of models for predicting global bathymetry, and we suggested future avenues of research to close that gap.

As we've stated, our original goal was to identify an ML model that performed well globally. We determined that no such model existed and developed the optimized regional ensemble for model selection. However, we note that the choice of region coverage size and shape was purely arbitrary. This approach gave us good results, which we've reported here, however, there are still a number of avenues we are interested in pursuing.

Our feature selection step landed on a set of features that were used with all models, although our work showed that each of the models keyed in on particular features. An interesting alternative would be to perform the feature selection step for each model independently. In addition to ensuring each model is given the most relevant features and removing noise that other models find useful, this would give us more insight by showing which feature and model combinations work well together.

We divided the world into a set of coverages based on computational expenses. We wanted to be able to run this experiment on commercial off-the-shelf hardware, and although it ran for several days, this was within our tolerance level. An alternative approach that would require significantly more resources would be to train on a sliding window. The coverages would overlap, and each grid cell would contribute to multiple coverages. Based on these results, we could define irregularly shaped regions around the best performing model.

We trained each coverage in isolation. This was useful for identifying which model performed best specifically for that coverage. It would be useful to see if a second-order model, one trained on all of the coverages that it performed best at, could further improve the results. The intuition here is that if a particular model key on a set of features, for example, fish biomass, then training on all regions where that feature dominates could increase the amount of training data available to that model.

Finally, we're interested in identifying regions that perform poorly for all models. We are interested in identifying regions where an investment in a detailed bathymetric survey would provide useful details that a model could then use to answer questions about other similar regions without requiring a survey in that area.

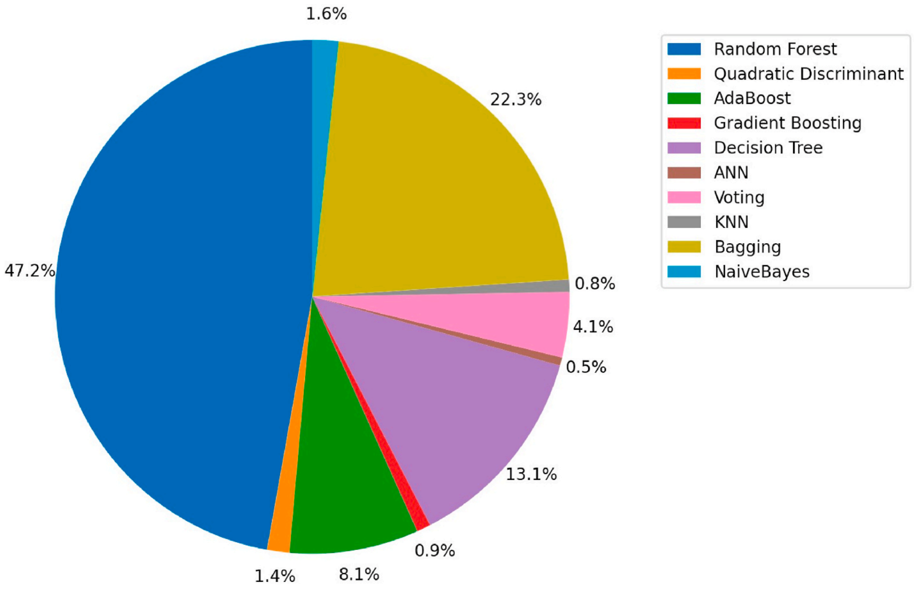

Fig. 7. Percentage of coverages where a model was “best fit.”

Table 3
Table of regression results.

Model	R^2 Score	Median Absolute Error	Mean Absolute Error
SVM Regression	0.841	365.23	480.21
Naïve Bayes	0.884	294.92	390.80
Linear Regression	0.885	290.13	387.48

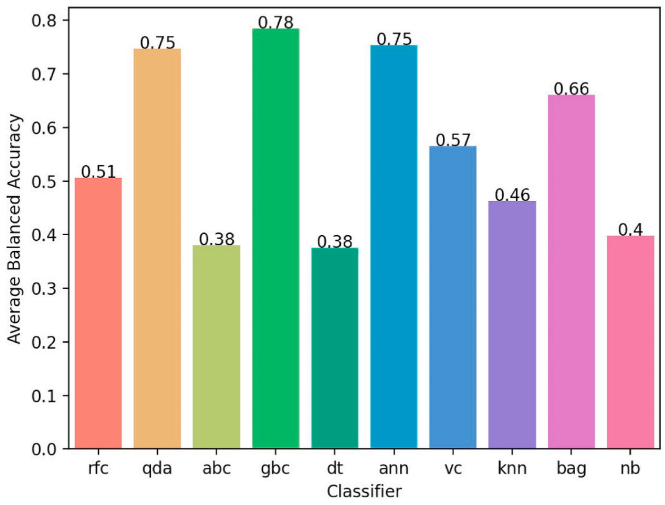

Fig. 8. Average prediction accuracy of coverages where a model performed well. GBC had the highest average accuracy but was only the best model for less than 1% of the coverages.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

7 Acknowledgement

MTH gratefully acknowledges the Louisiana Board of Regents through the Board of Regents Support Fund LEQSF (2016–19)-RD-B-07.

References

Agrafiotis, P., Karantzalos, K., Georgopoulos, A., Skarlatos, D., 2020. Correcting image refraction: towards accurate aerial image-based bathymetry mapping in shallow waters. Rem. Sens. 12 (2), 322.

Becker, J.J., et al., 2009. Global bathymetry and elevation data at 30 arc seconds resolution: SRTM30 PLUS. Mar. Geodes. 32 (4), 355–371.

Cao, B., Fang, Y., Jiang, Z., Gao, L., Hu, H., 2019. Shallow water bathymetry from WorldView-2 stereo imagery using two-media photogrammetry. Eur. J. Remote Sens. 52 (1), 506–521.

Chassignet, Eric P., et al., 2009. US GODAE: global ocean prediction with the HYbrid Coordinate Ocean Model (HYCOM). Oceanography 22 (2), 64–75.

Jena, Babula, et al., 2012. Prediction of bathymetry from satellite altimeter based gravity in the Arabian Sea: mapping of two unnamed deep seamounts. Int. J. Appl. Earth Obs. Geoinf. 16, 1–4.

Laske, Gabi, et al., 2013. Update on CRUST1. 0—a 1-degree global model of Earth's crust. Geophys. Res. Abstr. 15, 2658. EGU General Assembly Vienna, Austria.

Meissner, Thomas, Frank, J Wentz, Le Vine, David M., 2018. The salinity retrieval algorithms for the NASA Aquarius version 5 and SMAP version 3 releases. Rem. Sens. 10 (7), 1121.

National Geophysical Data Center, 1988. ETOPO-5 Bathymetry/topography Data.

Parekh, Payal, Follows, Michael J., Boyle, Edward A., 2005. Decoupling of iron and phosphate in the global ocean. Global Biogeochem. Cycles 19, 2.

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

Pedregosa, F., et al., 2011. Scikit-learn: machine learning in Python. J. Mach. Learn. Res. 12, 2825–2830.

Smith, Walter HF., Sandwell, David T., 1994. Bathymebibtric prediction from dense satellite altimetry and sparse shipboard bathymetry. J. Geophys. Res. Solid Earth 99 (B11), 21803–21824.

Smith, Walter HF., Sandwell, David T., 1997. Global sea floor topography from satellite altimetry and ship depth soundings. Science 277 (5334), 1956–1962.

Smith, Ryan N., et al., 2010. Planning and implementing trajectories for autonomous underwater vehicles to track evolving ocean processes based on predictions from a regional ocean model. Int. J. Robot Res. 29 (12), 1475–1497.

Tolman, Hendrik L., 2007. The 2007 release of WAVEWATCH III. In: Proc. 10th Int. Workshop of Wave Hindcasting and Forecasting.

Wei, Chih-Lin, et al., 2010. Global patterns and predictions of seafloor biomass using random forests. PLoS One 5, 12.

Yang, Jihoon, Honavar, Vasant, 1998. Feature subset selection using a genetic algorithm. In: Feature Extraction, Construction and Selection. Springer, pp. 117–136.