Fast Flood Model Error Correction using Tree Ensemble Methods

Project Category: Physical Sciences Application

Name: Raghav Sharma
SUNet ID: raghavsh
Department of Civil and Environmental Engineering
Stanford University
raghavsh@stanford.edu

Name: Minh-Tue Vo-Thanh
SUNet ID: minhtuev
Department of Computer Science
Stanford University
minhtuev@stanford.edu

Abstract

Simplified inundation models for modeling flood risk present a time-efficient alternative for expensive hydrodynamic models. Although orders of magbitude faster, these models are a modest representation of fluid-physics and have lower accuracies, necessitating a comprehensive error analysis. Benchmarking their performance with respect to a hydrodynamics model requires multiple runs of this expensive numerical model, making it difficult to scale the analysis for different topographies and storm levels. By using already compiled data from multiple runs of the hydrodynamic model, machine learning can be leveraged to predict errors given the physical, hydrologic, and topographic data of a domain, and use the predicted error to correct and augment the fast flood model results, thereby improving both accuracy and runtimes for inundation mapping. We have demonstrated the accuracy and effectiveness of this model, which can be applied in both research and industry.

1 Introduction

Climate Change has increased both the likelihood and intensity of extreme hydrologic events like coastal floods, which are increasingly exacerbated by rising sea levels. HighTide Intelligence is a climate intelligence startup from Stanford that has developed a holistic flood risk engine made possible by its proprietary fast flood model, adapted from [1].This fast flood model presents a modest representation of physical processes to map inundation levels given storm surge water levels, circumventing the use of traditional physics-based numerical Shallow Water Solvers and improving runtimes by orders of magnitude [2]. However, these superior runtime and lower computational costs come at the cost of reduced accuracy from simplied representation of the physics. Therefore it is critical to understand the limitations, uncertainties, and errors associated with these fast conceptual models with respect to the expensive, but more accurate hydrodynamic model [3]. In this project, we have investigated and developed an efficient method for evaluating uncertainty and error in the fast flood model without multiple runs of the CFD model for different water level forcings and geographies. The predicted error at each element of the domain can be used to correct the fast flood model results, thereby combining the use of Machine Learning and Simplified Flood Models into a Hybrid Modeling Framework to present a scalable and accuracte innundation model which is generalizable to different topographies, geographies, water levels, and storm approaches.

2 Related work

Climate science is an established and growing field, and generally relies on physical modelling of earth system processes. However, numerical modelling has runtime and computational disadvantages, which is why machine learning techniques are becoming increasingly widespread for predictive modelling of climate variables. Additionally, there has been a growing interest in Hybrid models

combining physical modelling with machine learning to augment the predictive ability of physical models [4].

[5] applies machine learning for empirical error correction of chaotic dynamic systems, which inspires our error correction methodology in a flood modelling context. We are inspired by this hybrid approach to extend and apply it to enhance the predictive ability of our simplified fast flood model with a goal of making it scalable and generalizable.

3 Data and Methods

3.1 Dataset Structure, Description, and Preprocessing

Each fast flood model and CFD model output is a 2D array of inundation depths due to storm surge of a given water level. An array of difference between the two models presents the discrepancies and error in the fast flood model which comes as a cost of improved efficiency and runtime. What interests us here is the difference/error in the fast flood model with respect to a baseline CFD model Delft3D flow, which will be the target variable in this problem. Inundation depth from flooding is dependent on domain-specific topographic and hydrologic features. Similar resolution rasters of physical/hydrologic features were converted to arrays to obtain a dataset of dependent and independent variables for each tile of the raster (element of a 2D array). The input features used are:

- Inland distance from the coastline - meters (int)
- Distance from the nearest water body - centimeters (int)
- Elevation with respect to North American Vertical Datum (NAVD 88) - meters (float)
- Water Level Forcing used in the flood models (storm surge height with respect to NAVD88) - meters (int)
- Flood depth output of the fast flood model - meters (float)

3.2 Data Preprocessing and Machine Learning Techniques

The initial task was to compile and preprocess a structured tabular dataset. The dependent variable was obtained by running both the models for 3 forcings and postprocessing the outputs using a GIS software. The 2D raster array of each input feature was flattened to create a structured tabular dataset of input features and target variable. Outliers and missing values were removed from the dataset and all the features were standardized by removing the mean and scaling by the variance. The final processed dataset has 20 million data points.

Although the uncertainty prediction task involves predicting a continuous value (the error), it was framed as a classification task by binning the target variable values which range from -10.3 to 10.9 into 100 uniform classes and tree ensemble methods were considered suitable for the multi-class classification job, given the tabular data. One reason for this is that more than the exact error value, we are interested in the error range. The hydrodynamic model depends to the many physical variables it takes as an input such as the viscosity, density, Manning's coefficient, etc and the baseline value against which we perform the error analysis is one iteration of the many permutations available which would give comparable if not the exact same result. The dataset was split into training, validation and test set with a 80:10:10 split. We decided to start with a Random Forest Classification (RFC) model and performed hyperparameter tuning using stratified 5-fold cross validation on accuracy score. To evaluate the performance of the prediction task in terms of actual error values, we inverse transform predicted binned class to a continuous error value, picked the middle bin value, and compared it with the continous target variable in the validation/test set. This interested us in also comparing how Random Forest Regression (RFR) would fare if we were to fix the above-mentioned physical variables for the baseline hydrodynamic model to predict the continuous error value instead of the error range. A similar approach to hyperparameter tuning was used for the RFR optimized for Mean Squared Error.

4 Experiments

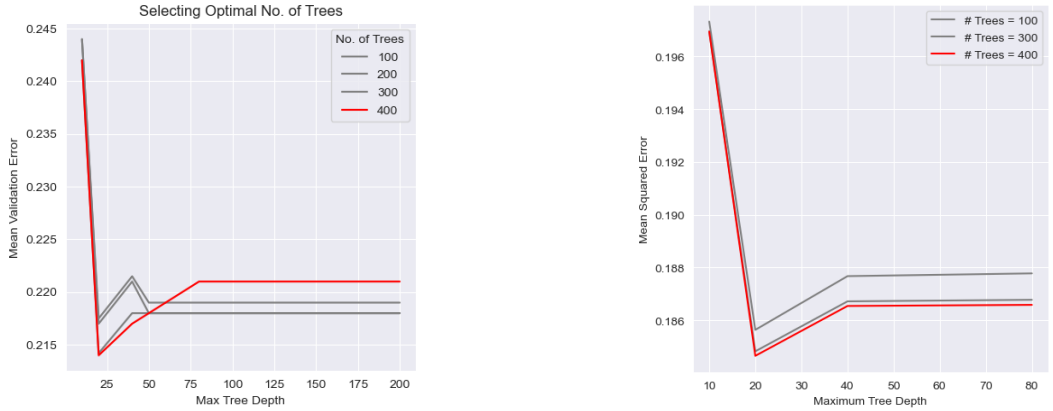
Hyperparameter tuning for the RFC and RFR was done for two hyperparameters: Number of trees and Maximum Tree Depth. For RFC and RFR, a 5-fold Sratified Grid Search Cross Validation (CV) was performed for tuning the hyperparamters. For selecting the optimal no. of trees and maximum tree depth, grid search was performed with the hyperparameter values presented in Table 1 and 2, respectively. After the optimal hyperparameters were determined, the RFC model was used to predict on the test set and the weighted precision, recall, F1-score, and accuracy score was noted. Similarly, the RFR model with the optimal hyperparameters was fitted on the entire training set and used to predict on the test set. The R^2 score and mean squared error on the test set were noted. We also compared how the midpoint of the predicted class (error range) from RFC would compare to the true continuous error value if we were to use classification to get an estimate of the continuous error instead of the range. We used traditional regression metrics such as Mean Squared Error and R^2 score to compare the continuous values. Figure 1a and 1b present the hyperparameter tuning results for RFC and RFR, respectively.

Table 1: RFC Hyperparameter values for CV

No. of Trees	100	200	300	400			
Maximum Tree Depth	10	20	40	80	100	150	200

Table 2: RFR Hyperparameter values for CV

No. of Trees	100	200	300	400
Maximum Tree Depth	10	20	40	80



(a) Random Forest Classifier

(b) Random Forest Regression

Figure 1: Hyperparameter Tuning

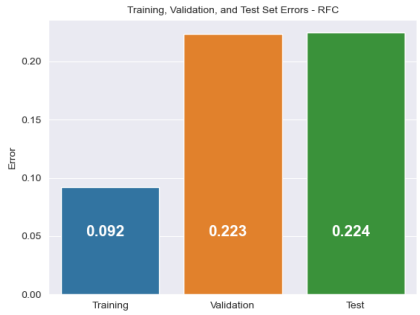
5 Results

Both the RFC and RFR turned out to have the same optimal hyperparameters. The optimal hyperparameters for the RFC and RFR, obtained through 5-fold Grid Search Cross Validation are:

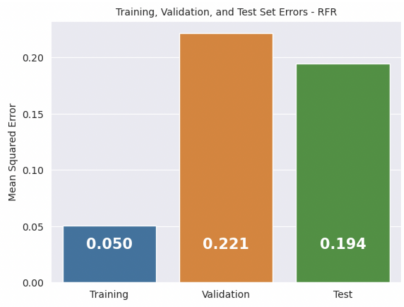
Number of trees = 20
Maximum Tree Depth = 400
Maximum Features for best split = Squareroot
Minimum number of samples required to split a node = 2

For the optimal hyperparamters, the classification metrics on the test set are presented in Table 3. The precision, recall, and F1 scores mentioned are weighted averages of the multiple classes.

On comparing the results of the inverse transformed class label (midpoint of the bin) to the true continuous error value in the validation/test set using regression metrics, we obtained an R^2 value of 0.905 and a mean squared error of 0.335. Figure 2 presents the training, validation, and tet set errors on the optimal RFC and RFR. Table 4 presents the test set metrics on the optimal RFR model fitted on the entire training set.



(a) Random Forest Classifier



(b) Random Forest Regression

Figure 2: Training, Testing, Validation set errors

Table 3: Classification Metrics on the Test Set

Accuracy	Precision	Recall	F1 score
0.78	0.77	0.78	0.77

Table 4: Regression Metrics on the Test Set

Mean Squared Error	R^2
0.194	0.945

6 Conclusion

Both the classification and regression models tend to do a fairly well job at predicting the error in the fast flood model given the input features and multiple water level forcings. With an accuracy of about 78% on the test set, the model generalizes fairly well, and can be used practically for conducting error analysis on the East coast of the United States, since the model was trained only on the Atlantic coast. However, given computational resource and time constraints, we were not able to achieve the intended accuracy levels. One limitation of the model is that it fails to predict accuractely on large negative errors. Negative error is where the fast flood model underpredicts. One reason for this is that the training set is imbalanced with a very tiny fraction of errors being negative since the fast flood model tends to overpredict exactly on the coastline. By adding these predicted errors to the fast flood model results, we can correct and enhance the predictive ability of a simplified model, helping it reach accuracy levels comparable to a much more sophisticated and expensive hydrodynamics model. This hybrid modelling approach can be extended to other earth sciences and climate sciences application by combining the advantages of physical and machine learning models.

7 Future Work

With a focus on tree ensemble methods, we intend to try out extreme gradient boosting (XGBoost) for the error prediction probelm. We also want to experiment with deep learning methods since they have been delivering impressive results in the past decade in computer vision. One particular approach considers the use of a Unet-like CNN architecture for image segmentation to classify 32x32 random crops of the inundation depth arrays into error classes. Since the raw dataset is in form of 2d arrays of input features representing the flooding domain, CNN-type architecture can prove to be useful for this problem. We will be exploring these techniques during the break and compare the results with the baseline Random Forest models.

Furthermore, we intend to scale this analyses by training the model on varied storm approaches, coastlines, and topographies, with a goal of generalizing the error prediction task for multiple domains across the United States and beyond.

8 Contribution

Raghav : Ran the CFD and fast flood model to obtain 2D array of each feature. Raster preprocessing using GIS. Model building and Hyperparameter Tuning on Random Forest Regressor and Classifier. Report Writing.

Minh-Tue : Data cleaning, exploratory analysis, and preprocessing. Model ideation and alternative strategies for problem formulation. Fitting Random Forest Regressor and Classifier on the full training set and predicting on the test set. Report Writing.

References

[1] J Teng, J Vaze, D Dutta, and S Marvanek. Rapid inundation modelling in large floodplains using lidar dem. *Water Resources Management*, 29(8):2619–2636, 2015.

[2] Jin Teng, Anthony J Jakeman, Jai Vaze, Barry FW Croke, Dushmanta Dutta, and SJEM Kim. Flood inundation modelling: A review of methods, recent advances and uncertainty analysis. *Environmental modelling & software*, 90:201–216, 2017.

[3] Delft3d - deltares. `https://oss.deltares.nl/web/delft3d/get-started`.

[4] Markus Reichstein, Gustau Camps-Valls, Bjorn Stevens, Martin Jung, Joachim Denzler, Nuno Carvalhais, et al. Deep learning and process understanding for data-driven earth system science. *Nature*, 566(7743):195–204, 2019.

[5] Peter AG Watson. Applying machine learning to improve simulations of a chaotic dynamical system using empirical error correction. *Journal of Advances in Modeling Earth Systems*, 11(5):1402–1417, 2019.