Fast Flood Model Error Correction Using Tree Ensemble Methods

Project Category : Physical Sciences Application

Raghav Sharma
Department of Civil and Environmental Engineering

Minh-Tue Vo-Thanh
Department of Computer Science

ABSTRACT

Simplified inundation models for modeling *flood risk* present a time-efficient alternative for the extensive hydrodynamic (CFD) models. Although orders of magnitude 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 the expensive CFD model, making it difficult to scale the analysis for different topographies and storm levels. By using already compiled data from multiple runs of the CFD 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 industry.

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 Stanford climate intelligence startup that has developed a holistic flood risk engine made possible by its proprietary fast flood model (Teng et al 2015). We want to evaluate the limitations, uncertainties, and errors associated with these fast conceptual models with respect to the expensive, but more accurate hydrodynamic models (such as Delft3D) that utilize Computational Fluid Dynamics (CFD). 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 conceptual presentation models into a Hybrid Modeling Framework to present a generalizable, scalable and accurate inundation model.

DATA AND METHODS

The 2D raster array of each input feature was flattened to create a structured tabular dataset of input features and target variable. 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. The input features are:

Inland Distance, Elevation, Distance from nearest water body, Fast Flood Model Depth, and Water Level Forcing.

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.

We divided the data into 80:10:10 splits and utilized 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 the predicted binned class to a continuous error value, picked the middle bin value, and compared it with the continuous target variable in the validation/test set. This interested us in also comparing how Random Forest Regression (RFR) would fare if we were 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.

EXPERIMENTS

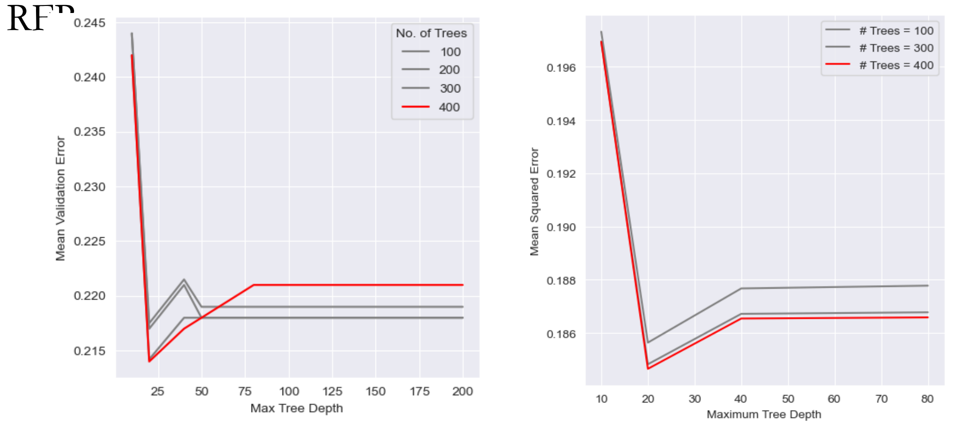
5-fold stratified cross validation (CV) was used for tuning the hyperparameters for both the RFC and RFR. Table 1 presents the hyperparameters tuned in the CV.

The experiments were performed on both local instances as well as on Google Cloud Platform.

Table 1: RFC and RFR Hyperparameter values for CV

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

Figure 1a and 1b present hyperparameter Tuning for RFC and RF



RESULTS

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

FUTURE WORK

We intend to experiment with XGBoost to perform regression/classification and note performance gains. Furthermore, we want to experiment with CNNs with a U-net-like image segmentation on 32x32 random crops of the 2D flood and feature maps.