Unraveling The Complexities of Urban Flood Hydraulics Through AI

Md Abdullah Al Mehedi (✉ mmehedi@villanova.edu)
   Villanova University

Virginia Smith
   Villanova University

Hossein Hosseiny
   Washington University in St. Louis

Xun Jiao
   Villanova University

Article

Keywords:

Posted Date: May 11th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1602023/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License. Read Full License
Abstract

As urbanization increases across the globe, urban flooding is an ever-pressing concern. Urban fluvial systems are highly complex, depending on a myriad of interacting variables. Numerous hydraulic models are available for analyzing urban flooding; however, meeting the demand of high spatial extension and finer discretization and solving the physics-based numerical equations are computationally expensive. Computational efforts increase drastically with an increase in model dimension and resolution, preventing current solutions from fully realizing the data revolution. In this research, we demonstrate the effectiveness of artificial intelligence (AI), in particular machine learning (ML) methods including the emerging deep learning (DL) to quantify urban flooding considering the lower part of Darby Creek, PA, USA. Training datasets comprise multiple geographic and urban hydraulic features (e.g., coordinates, elevation, water depth, flooded locations, discharge, average slope, and the impervious area within the contributing region, downstream distance from stormwater outfalls and dams). Classifiers such as logistic regression (LR), decision tree (DT), support vector machine (SVM), and K-nearest neighbors (KNN) are used to identify the flooded locations. Deep neural networks (DNN)-based regression model is used to quantify the water depth. The values of the evaluation matrices indicate satisfactory performance both for classifiers and DNN model (F-1 scores- 0.975, 0.991, 0.892, and 0.855 for binary classifiers; root mean squared error- 0.027 for DNN regression). This approach is a significant step towards resolving the complexities of urban flooding with a large multi-dimensional dataset in a highly computationally efficient manner.

Introduction

Flooding is globally one of the world's most destructive types of disasters. In the coming years floods are expected to be more frequent and larger globally\textsuperscript{1–3}. Coupled with the effects of rapid urban growth and climate change, the frequency of large flooding events are expected to increase, elevating the destructive impact floods\textsuperscript{4–9}. To untangle this challenge, engineers, planners, and emergency managers must be able to accurately anticipate flood extent and depth\textsuperscript{10}. Alteration to the depth and time of occurrence of precipitation as a result of climate change are forecasted to reshape the flooding scenarios encountered in many areas in the world shifting flood risk\textsuperscript{11}. These risks are also driven in part by local stormwater management and fluvial infrastructure, making predicting flooding a particularly arduous and critical challenge in the built environment\textsuperscript{12,13}. With the increase in urbanization, more impervious areas are generated resulting in less infiltration and greater flood peak and runoff\textsuperscript{14}. Hydrological response time is largely reduced in an urban setting increasing flood risk where the amount of impervious surface area is high\textsuperscript{15}. As a result, assessing flood risk in urban areas involves complex interaction between natural and engineered processes, some of which operate at very local scales, requiring fine resolution data\textsuperscript{16}. Numerous investigations have sought to define the pattern of hydrological regime transformation resulting from urban development\textsuperscript{17}. Alteration in urban river flow regimes is ascribed to the construction of impervious areas which facilitate rapid surface runoff from rainfall, the drainage of surface runoff through sewers to the river, and fluvial infrastructure\textsuperscript{18–20}. The proportion of urban land cover or the proportion of impervious cover within a catchment area provide predictor of changes in hydrograph characteristics, that lacks precision (e.g., the widely used Curve Number method)\textsuperscript{21,22}. Hydraulic models provide more precise results, demand expensive computational results and data.

The complex and ever-changing urban landscape make urban flood prediction and modeling computational expensive and often infeasible due to high resolution data requirements. Numerical analysis of hydraulic equations across spatio-temporal boundaries can be increasingly expensive depending on the resolution needed. In addition to the complex non-linear relationship among the features of urban flooding, estimating hydraulic parameters using physics-based equations is computationally expensive, as this process requires a large amount of memory allocation\textsuperscript{23}. Due to the computational expense, the range of input data is often limited, resulting in models at course resolutions, exclusive to different types of hydraulic conditions and potentially relevant parameters. Moreover, uncertainty of the parameters, defective model
calibration and errors in the measurements can serve as an accelerator to the computational expense. Answering the challenge of urban flood necessitates models that can efficiently and effectively represent flood extent with available data, in a quick and robust manner.

Data-driven prediction with Machine Learning (ML) techniques in the field of artificial intelligence (AI) provides a potential solution. ML is rapidly growing in popularity across many fields. ML methods, including the emerging deep learning (DL) methods, have been successfully applied to the field of water resources for stage-discharge (Q/h) relationships, rainfall-runoff, sediment transport, flood prediction, water quality analysis. AI models are specifically convenient when the uncertainty in model parameters, complexities in the physics-based equations and computational efforts are significantly high, such as in urban hydrology. To predict the flooded locations, a set of classifiers e.g., logistic Regression (LR), support vector machine (SVM), K-nearest neighbor (KNN) and decision tree (DT) are used. Binary classification generates the output in the form binary data i.e., 0s and 1s, which represent whether a location is flooded or not in the study area. LR model has shown satisfactory performance in the previous study in classifying flooded locations using the matrix of probability of detection on average for flood events. The DT algorithm showed good performance with Minimum Absolute Error and classification accuracy in IoT (Internet of Things) based flood detection and notification system. The SVM attributes non-linear transformation of geographic and hydraulic features in higher dimensional feature space. Highly satisfactory performance was achieved from SVM algorithm in detecting the area of prone to flood risk in the river basin of Buzau in Romania. Supervised regression with Deep Neural Network (DNN) is performed to predict the water depth within the model domain. Real-valued regression with artificial neural network provides a reliable means of predicting the flooding depth with a good performance range. In this investigation, multilayer perceptron-based feed forward neural network with back propagation algorithm is used to perform the regression task i.e., predict water depth in the computational space. DNN regression model is specifically suitable in predicting the flooded depth with representative geographic and hydraulic features. To increase the efficiency of entire process of flood prediction, an efficient ML workflow plays a vital role by minimizing human involvement and increasing automation through coding.

This study delineates a novel data-driven strategy towards unravelling the complexities in urban flooding environment using multiple AI approaches i.e., a set of binary classifiers to detect flooded locations and DNN regression to predict water depth. Multiple geographic and urban hydraulic features are used to prepare the flood map with locations and magnitudes. The approach incorporates the characteristics of the urban area by introducing urban hydraulic drivers in training the models for prediction. The outcomes of this research have significant potential to advance the flood preparedness mechanism for urban areas vulnerable to the flash floods where devastation due to rapid accumulation of flood water is significantly high. The quick and flexible model presented here is transferable and can be utilized to prepare large-scale flood maps in an inexpensive and efficient way in the cloud-computing platform across urban areas. The approach outlined in this study has the potential to efficiently predict urban fluvial flooding for a range of scenarios.

**Data And Method**

**Study Area**

The study area considered in this study is the lower part of the Darby Creek (DC), along the southwest boarder of Philadelphia, PA, USA showed in the Fig. 1. The alluvial channel of the Creek flows through a floodplain with fully urbanized settings which is subject to frequent flooding. The population residing near to the creek are subject to flood significantly. The portion of the river considered in this study flows from the Mt. Moriah Cemetery (upstream) to the confluence with the Delaware River (downstream) and carries alluvial deposits through an urbanized setting, approximately 15 river kilometers (rkm). Darby Creek plays an important role to the adjacent environment and ecology; it is also flood prone area. It also offers a unique environment for various plant and animal species.
Preparing Hydraulic Dataset in iRIC

Hydraulic models are simulated in the iRIC platform to generate dataset for ML classifiers and DNN regression model. The iRIC is a numerical tool capable of modelling rainfall runoff generation, flooding, and sediment dynamics. It receives terrain and hydraulic data (e.g., water surface elevation, roughness) for the model calibration purpose. FaSTMECH (Flow and Sediment Transport with Morphological Evolution of Channel) is used as a solver in this study\textsuperscript{50}. The terrain data is discretized to a size of 5 m\(^2\) for every computational cell. As the higher discharges from the upstream side of the river are responsible to the morphological changes, higher discharge values from the highest flood event in Darby Creek are chosen to create scenarios for AI models. Multiple scenarios are created using various constant discharge values upstream of DC within a certain range. The discharge data from observed flood events in the time span of 14th July to 16th September is obtained from USGS peak stream flow data (USGS gage 01475548)\textsuperscript{51}. A set of discharge values is chosen to execute Machine Learning and Deep Learning algorithms used in this study. The discharge values are 37, 42, 45, 50, 52, 61, 83, 95, 99 and 164 m\(^3\) per second (cms). The outcomes generated by the iRIC are water surface elevation and flooding depth. A set of urban hydraulic features i.e., the quantity of the impervious areas within a contributing area to a specific location, downstream distance from the hydraulic structures e.g., stormwater outfall and dam are introduced in this study to integrate the effect of urban attributes with the flooding extent and magnitude. Furthermore, average slope of the contributing area is derived through GIS analysis and incorporated to represent the flow accumulation to a specific location. Hydraulic model calibration requires elevation data from the floodplain and bathymetry of the channel. Water surface elevation at the upstream of USGS Cobb Creek gage at Mt. Moriah cemetery for the flooding event of 30th August of 2009 is utilized to calibrate the hydraulic model\textsuperscript{43}.

Al Models

The quantification of the flood extent and depth by the ML framework was tackled in three steps. Firstly, an exploratory analysis and feature engineering is performed to study and transform the entire dataset prepared by multiple geographic and hydraulic features, listed in Table 1. After analyzing dataset and conducting necessary transformation on the features, classifiers, such as Logistic regression (LR), K-nearest neighbors (KNN), decision trees (DT), support vector machines (SVM), are trained using the data prepared in the first step to locate or classify the flooded locations for each scenario of various upstream discharges. Third, an artificial DNN is used to prepare a regression model to predict the depth of water within the computational domain. ML classifiers and neural networks-based regression models are evaluated using several error matrices, e.g., F1-score, Jaccard similarity score and Root Mean Square Error (RMSE). The algorithms are tuned and optimized by altering the hyperparameters to reduce the error and obtain satisfactory performance. The ML workflow of flood prediction is described in Fig. 2. The entire process can be divided into groups of tasks, i.e., data collection, exploratory data analysis, feature engineering, model training, model evaluation, model deployment, and model improvement. Details are provided in the following sections. The steps are further categorized into their distinct group namely transformer, estimator, and evaluator.
Table 1
Full descriptions of the predictor and target variables used to train/test the ML classifiers and DNN regression model.

<table>
<thead>
<tr>
<th>Features</th>
<th>Full descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_1$</td>
<td>x-coordinates of every location in the model domain</td>
</tr>
<tr>
<td>$x_2$</td>
<td>y-coordinates of the same location</td>
</tr>
<tr>
<td>$x_3$</td>
<td>Elevation in meters of same location</td>
</tr>
<tr>
<td>$x_4/y_2$</td>
<td>Depth of water in meters</td>
</tr>
<tr>
<td>$x_5/y_1$</td>
<td>Flooded locations</td>
</tr>
<tr>
<td>$x_6$</td>
<td>Average slope of the contributing area of every point in percentage</td>
</tr>
<tr>
<td>$x_7$</td>
<td>Number of impervious locations of the contributing area</td>
</tr>
<tr>
<td>$x_8$</td>
<td>Downstream distance from the stormwater outfalls</td>
</tr>
<tr>
<td>$x_9$</td>
<td>Downstream distance from the dams</td>
</tr>
<tr>
<td>$x_{10}$</td>
<td>Upstream river discharge in m$^3$/s</td>
</tr>
</tbody>
</table>

Feature Engineering

Scikit-learn is used as the ML library for feature engineering in Python\textsuperscript{52}. It offers several classification, regression and clustering algorithms including LR, KNN, DT and SVM which are used as binary classifiers for identifying flooding locations in this study. Modules needed for ML and Deep Learning algorithms such as optimization, linear algebra, integration, interpolation, special functions can be accessed through SciPy\textsuperscript{41}. Independent variables for Binary Classifiers and DNN regression model are listed in the Table 1. Flooded location is used as the target variable denoted by $y_1$ in case binary classifiers and water depth, $y_2$ as a target variable in case of DNN model. Spatial information, coordinates and elevation values are obtained from the original Digital Elevation Model (DEM) of the study area using ArcGIS Pro. Water Depth and Discharge values are extracted through simulating multiple hydraulic models in iRIC platform. Average Slope and number of impervious cells of the contributing area of every point of the DEM are urban hydraulic features, which have not been introduced before as a training feature for AI models. ArcPy, a Python site package that offers an effective and efficient way to perform geographic data analysis, data conversion, data management, and map automation using Python was utilized to generate the contributing areas of every cell upstream in the model domain\textsuperscript{53}. It can be compared with the upstream area contributing to those cells. No modification was needed to alter the data type, as it is generated from iRIC-FaSTMECH simply as binary data type. The main data frame is constructed through concatenating datasets derived from different upstream discharges (Q) scenarios.

Before initiating the learning process, the feature importance was analyzed. Feature Engineering tasks used in this study to prepare the datasets for the ML/DL algorithms include numerical imputation, outlier detection with standard deviation and dropping, splitting training/testing datasets, and scaling with normalization. The proportion of the train-test split is assumed to be 80/20 for both ML classifiers and DNN regression. In the Eq. 1 how the normalization of the features performed can be observed. $X$ denotes the feature vector including all the features used to train/test the models. Preparation of dataset for training the DNN is identical to the preparation of the training dataset for ML Classifiers. Eighty percent (80%) of the data is used to train and rest of the data is used to test both the ML classifiers and DNN regression model.
Identifying Flooded Locations with ML Classifiers

Logistic Regression (LR)

Linear regression searches a function that builds a relationship to a continuous dependent feature/variable, $y$, to some outcome/predictors (independent features $x_1, x_2$, etc.). LR is a variation of linear regression, utilized when the existing dependent variable/outcome, $y_1$, is categorical. It generates a formula that forecasts the probability of the category as a function of the independent features. Logistic regression fits a special s-shaped curve (sigmoid function) by taking the linear regression and converting the numeric into a probability with the function, which is known as the sigmoid function $\sigma$.

$$h_\theta(x) = \sigma\left(\theta^T x\right) = \frac{e^{\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ldots}}{1 + e^{\theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ldots}}$$

The probability of a category 1 (a location being flooded) = $P(Y = 1|X) = \sigma\left(\theta^T x\right) = \frac{e^{\theta^T x}}{1 + e^{\theta^T x}}$. Therefore, LR passes the features (e.g., $x_1 =$ elevation, $x_2 =$ slope of the contributing area, $x_3 =$ water depth, etc.) through the logistic/sigmoid functions; however, considers the outcome as a probability. The goal of LR algorithm is to identify the best parameters $\theta$, for $h_\theta(x) = \sigma(\theta^T X)$, in such a way that the algorithm forecasts a cell is being flooded or not in the model domain.

Decision Tree (DT)

Decision tree learning is one of the predictive modeling approaches used in statistics, data mining and machine learning. It uses a decision tree (as a predictive model) to go from observations about an item e.g., features mentioned in the Table 1 (represented in the branches) to conclusions about the item's target value e.g., binary decision on a location being flooded or not (represented in the leaves). From scikit-learn, Decision Tree Classifier is used to perform the classification task on flooding location.

Support Vector Machine (SVM)

SVM works by mapping data to a high-dimensional feature space so that data points can be categorized, even when the data are not otherwise linearly separable. A separator between the categories is found, then the data is transformed in such a way that the separator could be drawn as a hyperplane. Following this, characteristics of new data can be used to predict the group to which a new record should belong.

K Nearest Neighbors (KNN)

The principle of KNN is based on the concept that the k closest objects or similar cases in the p-dimensional space (the number of dimensions is identical to the number of the features mentioned in the Table 1) determine the class of an unknown variable i.e., flooded locations. KNN aims to partition n observations (number of rows in the flood prediction data frame) into k clusters tagging each observation (rows in the dataframe) to a specific cluster with the cluster centers or cluster centroid or the nearest mean serving as a prototype of the cluster. The entire data space is partitioned into Voronoi cells in this approach. When features are obtained in different physical units with vastly varying scale, normalizing the training features and outcomes can improve the accuracy of the KNN algorithm as it depends on distance of the data points for the classification.
Predicting Flood Depth With Deep Neural Network

The goal of using DNN regression model is to construct a model to predict the water depth ($y_2$) using existing features derived from the hydraulic model and GIS data. To do this, a full set of multiple hydraulic variables/features mentioned in the Table 1 is used to train/test the DNN model. Scikit-learn is used as the ML library for feature engineering using Python\textsuperscript{41}. It offers several classification, regression and clustering algorithms including LR, KNN, DT and SVM which are used as binary classifiers for identifying flooding locations in this study. Modules needed for ML and Deep Learning algorithms such as optimization, linear algebra, integration, interpolation, special functions can be accessed through SciPy\textsuperscript{42}. Independent variables for Binary Classifiers and DNN regression model are listed in the Table 1 and fed into the DNN. Open-source library TensorFlow is used in this study work to construct DNN model as it has an excellent particular focus on the inference and training of DNN\textsuperscript{48}. Training a model with TensorFlow Keras typically starts by defining the model architecture.

Input layer contains features denoted by $x_i$ in general which is similar to the binary classification problem. The weights imposed on different features, aggregation of multiple features, further weights before the output layer and the activation functions are denoted with $W$, $z_i$, and $a$ respectively. Finally, target variables (water depth) are generated from output layers. In Fig. 3 (a), it can be observed that introducing neural networks improves the prediction performance significantly through the introduction of non-linearity among the input and target features. The activation function used to introduce the non-linearity to the model is ReLU (rectified linear unit) function showed in the Fig. 3 (b). This function returns the standard ReLU activation: maximum ($x$, 0), the element-wise maximum of input tensor ($x$) and 0 with default values. Total number of layers used to perform DNN is four including a normalized input feature layer, two hidden layers and a linear single-output layer. Total number of weights for each trainable neuron is 4,609 where 11 neurons are found to be non-trainable.

Urban hydraulic feature importance is studied by analyzing the sensitivity of the change in feature values over the target variable, water depth and Permutation Feature Importance (PFI) technique in the computational domain. The values of impervious area, average slope of the contributing, downstream distance (DD) from the Stormwater Outfall (SO) and Dams (DO) are varied (5%, 10% and 20%) to observe the impact on the target variable in the DNN regression Model. The RMSE values are obtained from the difference between the series of the target variable, water depth after running the DNN model with the changed features and the series before running the model. In the PFI technique, DNN model is run with the values of a specific feature e.g., impervious areas of the contributing area permuted/shuffled keeping the other features constant and the change in the RMSE values are recorded\textsuperscript{41}.

Results And Discussion

Scenarios from Hydraulic Model

The relationship with the water depth in the computational space and river discharge is highly non-linear. High variation in the geometry of the channel and roughness of both channel and floodplain against the flow can enhance the non-linearity in the system. In this section, water depth variations with their corresponding locations are presented with respect to multiple scenarios with several upstream discharge values, including 37, 42, 45, 50, 52, 61, 83, 95, 99 and 164 m$^3$ per second (cms). The validated hydraulic model\textsuperscript{49} from iRIC is used to simulate and create scenarios having the water surface elevation and depth, locations, and binary output regarding a certain location is flooded or not as results. In the Fig. 4, two plots of water depth with the locations for the scenarios with discharges 52 and 99 cms can be observed which are obtained from hydraulic simulation in iRIC.

Binary Flood Map
The features used to train the ML classifiers (LR, DT, SVM and KNN) are \( X = [x_1: \text{x-coordinate}, x_2: \text{y-coordinate}, x_3: \text{ground elevation}, x_4: \text{water depth}, x_5: \text{average slope}, x_6: \text{number of impervious locations}, x_7: \text{downstream distance from SO}, x_8: \text{downstream distance from dams}, x_{10}: \text{upstream discharge}] \). Locations are classified using their corresponding water depth into two classes, whether a particular location is flooded or not (1/0). The Fig. 5 illustrates the distribution of flooded location for the scenarios with the upstream discharge values of 52 and 99 cms predicted with the DT algorithm. DT is the top performer among all other ML classifiers with the F1-socre and Jaccard Similarity matrix of 0.991 and 0.966. Several conventional statistical measures are available to evaluate the performance of the ML classifiers. Mean Absolute Error (MAE), F1-score, True Positive (TP), False Negative (FN), Root Mean Square Error (RMSE) Jaccard similarity score, log-loss are among the popular choices and provide evaluation of the models in quantitative terms\(^57\). In this study, F1-score and Jaccard similarity score are used to evaluate the ML classifiers. The F1-score is used to evaluate binary classification algorithms, e.g., logistic regression, which generates binary outputs of whether a location is flooded or not. The harmonic mean of the model’s precision and recall is calculated to determine the F1-score\(^58\). The performance of the ML classifiers can also be determined from the confusion matrix, shown in the Fig. 5. A confusion matrix is a table that is used to define the performance of a classification algorithm. A confusion matrix visualizes and summarizes the performance of a classification algorithm. It is used to visualize the performance of a classifier, typically a supervised classification algorithm\(^59\). Two parameters needed to estimate the F1-score are precision and recall. Precision represents the fraction of the number of instances which the model correctly predicted \((T_p)\) and the sum of all instances that are incorrectly predicted as true \((F_p)\). Recall, sometimes referred to as sensitivity, is the fraction of the number of instances which the model correctly predicted \((T_p)\) and the sum of all instances that are incorrectly predicted as false \((F_n)\)\(^60\).

\[
2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} = \frac{T_p}{T_p + 0.5(F_p + F_n)} \tag{3}
\]

The Jaccard coefficient quantifies similarity between finite sample sets and is determined as the size of the intersection divided by the value of the union of the sample sets. Given forecasted values of flooding occurrence as \(\hat{y}\) and actual values of flooding occurrence as \(y\), the Jaccard index can be defined as

\[
\text{j}(y, \hat{y}) = \frac{y \cap \hat{y}}{y \cup \hat{y}} \tag{4}
\]

All ML classifiers conveyed satisfactory performance in isolating flooded locations as the values of the error matrices in the Table 2 are closer to the unity. From the Table 2, it can be observed that LR and DT outperform other classifiers. The F1-score and Jaccard similarity matrix of the LR and DT are 0.975, 0.991 and 0.995, 0.986 respectively which are greater than the values of the SVM and KNN.

<table>
<thead>
<tr>
<th>Binary Classifiers</th>
<th>F1-score</th>
<th>Jaccard Similarity Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logistic regression (LR)</td>
<td>0.975</td>
<td>0.995</td>
</tr>
<tr>
<td>Decision tree (DT)</td>
<td>0.991</td>
<td>0.966</td>
</tr>
<tr>
<td>Support vector machine (SVM)</td>
<td>0.892</td>
<td>0.901</td>
</tr>
<tr>
<td>K-nearest neighbors (KNN)</td>
<td>0.855</td>
<td>0.810</td>
</tr>
</tbody>
</table>

Performance of the binary classifiers can also be illustrated in the form of confusion matrix, comparing correctly predicted outcomes with the incorrectly predicted outcomes. In the Fig. 6, the confusion matrices are showed to illustrate the performance of the ML classifiers. Number of flooded cells predicted correctly by LR algorithm (a) numbers 8332 (96.8%),

Page 8/21
while incorrectly predicted cell count is 324 which is significantly lower than the number of correctly predicted locations. Similarly, the number of correctly predicted not-flooded cells count 54,747 (99%) where the number of incorrectly predicted not-flooded cells are 47. A total of 96.8% of predicted cells are correct, suggesting a highly accurate model performance.

**DNN Regression to Predict Flood Depth**

Artificial neural network with a single hidden layer is not capable of extracting the insights of the non-linearity and complexity of the flood prediction. Therefore, DNN with multiple hidden layers is incorporated in the prediction process of water depth. Adding more hidden layers increases the accuracy of prediction. However, inclusion of a large number of hidden layers requires high computational power and may result in overfitting the model.\textsuperscript{61,62} From the model evaluation, it can be observed that the DNN described in this manuscript reflected the complexities of river flood prediction. To capture the high amount of non-linearity among the geographic and urban hydraulic features mentioned in this paper and establish linkage among them, it is a prerequisite to introduce multiple hidden layers. Hidden layers with the nodes built in them are used to train the model through an iterative optimization process. A total of three hidden layers is used with 64 neurons assigned to each. While the number of epochs found best with a minimum error is 110. 80 percent of the whole dataset was used for model training purposes, while 20 percent was used for testing the performance of the DNN. The activation function for hidden layers used is ReLU. Other popular activation functions such as hyperbolic tangent, sigmoidal or leaky ReLU functions are recommended to introduce the non-linearity in DNN. The model evaluation matrix, RMSE value of 0.027 illustrates the DNN regression algorithm conveyed satisfactory performance in resolving the high non-linearity in the flooding depth prediction process.

Urban hydraulic features, i.e., average slope and the number of impervious cells of the contributing area, are introduced in this process to train the DNN regression model. It is clear from the Fig. 7 that the flood depth is highly correlated and sensitive to the upstream discharge. With the increase in the upstream discharge, flood depth also increases. In the DNN regression model training phase, a non-linear correlation is built among these urban hydraulic features and flooding amount and extent. By introducing these urban hydraulic features, connections among the features and target variables i.e., number of impervious locations and water depth are established. The difference maps in the Fig. 8 illustrate the variation in the predicted water depth from DNN regression model and hydraulic model. The spatial distribution of the differences is not significant (in the scale of 0.01 m) for varying upstream discharge condition. Based on the error matrices and difference mapping, it can be concluded that the performance of the DNN regression model is excellent for the urban hydraulic features considered in this study.

**Urban Hydraulic Features Importance**

Permutation Feature Importance (PFI) and sensitivity analysis approach are used to determine the impact of the urban features on the DNN regression prediction output i.e., water depth based on the RMSE value as the indicator. PFI measures the variation in the prediction error of the model after the feature's values are permuted.\textsuperscript{63} This approach quantifies the change in the RMSE values as the prediction error after a series of feature values of interest is permuted/shuffled breaking the linkage between the feature (e.g., downstream distance from stormwater outfall) and target variable (e.g., water depth). This measure is an indicative of the dependency of the model outcome to a specific feature. A sensitivity analysis for the increase in the feature values (5%, 10% and 20%) is performed to quantify the response of the target variable in the DNN regression model, i.e., water depth to the variation in the urban features using RMSE value. From both analysis, impervious area is found to have the highest importance to predicting flood depth compared to the other features. This is logical, as impervious area directly contributes to runoff and hence to the accumulation of water. The importance scores in both approaches are shown the Fig. 9 to illustrate the significant response of water depth predicted from the DNN regression model due to the change in the urban hydraulic features. Impervious areas showed the highest influence over the target variable i.e., the water depth predicted from the DNN regression model followed by the downstream distance of the
stormwater outfall and dams and average slope to the contributing area in case of PFI and 20% increase in the feature values. For the 5% and 10% increase/decrease in the feature values, downstream distance from the stormwater outfall showed highest impact. The score of the feature importance, the RMSE value increases for all features with the increase in the change of the feature values from 5–20% shown in Fig. 9. Highest response in the RMSE value can be observed in the case of PFI where the feature values are permuted instead of adjusted with a simple sensitivity analysis by percent increase.

**Conclusion**

Robust and real-time prediction of the flooding is critical to alleviate the growing risk of urban flooding. Estimation of water depth of the river and floodplain for various scenarios is of paramount importance in urban flooding planning and management, particularly as many municipalities seek to install or upgrade infrastructure. Data-driven ML approaches provide a path to circumvent the complexities of urban flooding using geographic and urban features outlined in this paper has the potential to get insights of the flooding attributes. Highly expensive and inefficient computationally using physic-based numerical models become burdensome at the city-scale and beyond. Traditional hydraulic models depend on solving physics-based differential equations, they require extraordinarily processing power and high memory allocation, specifically for large amount of data and thus perform much slower compared to data-driven methods presented in this paper. In this study, flooding in urban areas, such as the highly urban Darby Creek watershed, is predicted using hybrid physics-informed data-driven techniques. A novel approach to classify and predict the flooded locations and depth using various ML classifiers and DNN-based regression method illustrates a promising ground and potential to entirely shift into the data-driven techniques. Derived urban hydraulic features, i.e., impervious locations and average slope within the contributing area, downstream distance from the stormwater outfall and dams, are introduced in this paper to incorporate the unique impact of urban features on the flooding extent and magnitude which was not present in the previous research works. Future inclusion of additional parameters and resolutions can aid in deepening the understanding of urban hydrology.

A set of binary classifiers (LR, DT, SVM, KNN) are used to identify the flooded locations and DNN regression model with multiple hidden layers is applied to capture the high non-linearity and quantify the flood magnitude in an urban environment. Both the classification and regression algorithms trained to predict the flooding locations and depth in urban areas with minimum error generated satisfactory outcomes. All error matrices used to evaluate the performance of the binary classifiers are F1-score, Jaccard Similarity matrix and confusion matrix delineate promising capability of the ML classifiers in isolating flooded locations. RMSE value used to evaluate the adequacy of the DNN algorithm in predicting water depth. also showed satisfactory performance for the unique datasets with geographic, derived urban and physics-informed hydraulic features. Further, the urban hydraulic feature importance scheme quantified the impact of urban features over the outcome (water depth) of DNN regression model. Therefore, the satisfactory performance of the proposed framework presented here shows a higher potential for flood prediction in an urban environment, by accounting for the influence of urban features compared to the traditional physics-based hydraulic models.

The performance of the ML classifiers and DNN regression models can be improved with the increase in the discretization of the computational domain creating more training and testing data. Geographic and hydraulic features can be stored in the web where the entire training or testing workflow is possible to be executed in the cloud-computing platform. Further, other machine learning and deep learning classifiers and regression models such as Gaussian Process classification, Bayesian classification, Histogram-based gradient boosting, and Long Short-Term Memory regression can be studied with the river hydraulic dataset. Notably the transferability of this method is data limited. Linking to physical models has the potential to advance model capabilities, as well as allow for deeper insight into urban hydrologic processes. Future work in this area is highly recommended as the data availability and computational power are increasing rapidly. The approach
outlined in this study has a potent to be combined with the weather forecast models paving the way of feasible and inexpensive quantification of real-time flooding scenarios.

**Declarations**

**Acknowledgements**

This work was supported by the Villanova Center of Resilient Water System (VCRWS) of Villanova University.

**Competing interests**

The authors declare no competing interests.

**Funding**

Department of Civil and Environmental Engineering

**Data Availability**

Data collected for the study can be made available upon request from the corresponding author.

**Corresponding author**

Correspondence and requests for materials should be addressed to Md Abdullah Al Mehedi

**References**


48. 16125.pdf.


Figures
Figure 1

The figure above shows the global (a) and local (b) terrain of the study area. Maps are processed and generated in the ArcGIS Pro platform\textsuperscript{49}.

Figure 2

Flood Prediction workflow using machine learning classifiers and neural network regression techniques illustrates how the tasks are linked from the data preprocessing steps to the model deployment and maintenance stage.
Figure 3

DNN architecture for flood prediction (a). A perceptron with summation of weights, features, and biases with the activation function (b). Input layers consist of all the input features and their corresponding weights. Aggregation of the features, weights and biases constitutes output layer.
**Figure 4**

Flooding extent and depth estimated by the FaSTMECH solver in the iRIC platform for the scenarios with discharge values a) 52 and b) 99 cms. These scenarios are used to train/test ML classifiers and DNN regression algorithm. Figures are created in ArcGIS platform after importing iRIC result file.

**Figure 5**

Flooded locations detected by the DT algorithm for representative scenarios with upstream discharge values (a) 52 and (b) 99 cms. Prediction performance of the DT algorithm is the best among all other classifiers.
Confusion Matrix can be used to observe the performance of ML classifiers i.e., logistic regression (a), decision tree (b), support vector machine (c) and K-nearest neighbors (d). Total number of predicted cells from ML classifiers are compared to the actual values from hydraulic model to extract the number of correctly predicted cells.
Figure 7

Predicted water depths for the representative scenarios covering the full range of upstream discharge values (a) 42, (b) 52, (c) 99 and (d) 164 cms outputs from the DNN regression model.
Figure 8

Error distributions calculated by subtracting the water depth from DNN regression model output and iRIC simulation is shown here for two scenarios with upstream discharge of 52 (a) and 99 (b) cms. Maps are processed and generated in the ArcGIS Pro platform.
Figure 9

Change in the RMSE value of the DNN regression algorithm due to the % increase/decrease in the feature values and Permutation Feature Importance.