Article

Using Machine Learning to Improve Performance of a Low-Cost Real-Time Stormwater Control Measure

Marzieh Khosravi[a], Md Abdullah Al Mehedi[a], Sara Baghalian[b], Michael Burns[a], Andrea L. Welker[c], Michael Golub[d]

[a]PhD Candidate, Department of Civil and Environmental Engineering, Villanova University, 800 Lancaster Ave, Villanova, PA 19085, USA; E-mail: mkhosrav@villanova.edu, mmehedi@villanova.edu, mburns35@villanova.edu
[b]Postdoctoral Researcher, Department of Civil and Environmental Engineering, Villanova University, 800 Lancaster Ave, Villanova, PA 19085, USA; USA; sara.baghalian@villanova.edu
[c]Professor and Dean, School of Engineering, The College of New Jersey, PO Box 7718, Ewing, NJ 08628, USA; welkera@tcnj.edu
[d]Director of Global Data Science Research, Innovation and Operations, Merck; 4golub@gmail.com

Abstract: The alteration of natural land cover to impervious surfaces during development increases stormwater runoff. Stormwater Control Measures (SCMs) are used to manage water quantity and enhance water quality by restoring the hydrologic cycle altered by development. Often, SCMs have an outflow pipe to handle overflows or to manage the release of water detained when infiltration is not possible. Traditionally, these are static controls (e.g. a small orifice is used to restrict the volume of outflow), however, these systems can be improved by instituting real-time controls (RTC). RTC improve the functionality of SCMs by dynamically controlling outflows to adjust to environmental conditions. A major impediment to the widespread implementation of RTC is the high cost of installation and operation. This study utilized machine learning methods to develop a forecasting approach for the implementation of low-cost RTC that were implemented on a programmable gate of the outlet structure of a multi-stage basin in southeastern Pennsylvania. The goals were to decrease the peak flow exiting the basin during rain events, increase the volume of water detained, decrease the number of overtopping events, maintain healthy vegetation in the basin, and protect the downstream vegetation from erosion. Multiple popular data science algorithms were evaluated including multiple linear regression and long short-term memory. These algorithms were used with a dataset, which consisted of four years of historical sensor data, collected in 5-minute intervals, to train models to predict water levels to optimize operations. The accuracy of 30 models with three different methods of handling missing values were compared. A long short-term memory model configured with a 30-minute lead-time produced the best results. Having an approximate same lag time of 30 minutes for the contributing drainage area of the SCM provided a sufficient RTC functioning period to improve the performance of the outlet structure.

Highlights

- Real-time controls (RTC) can improve stormwater basin performance.
- The high cost of traditional RTC has inhibited widespread adoption.
- Low-cost RTC were added to a multi-stage stormwater basin to improve performance.
- The RTC were optimized using machine learning.

Keywords: Real-Time; Stormwater; Control Measure; Low-Cost; Machine Learning; Time-series; LSTM

1. Introduction

Development results in the transformation of pervious land cover to impervious surfaces, which triggers increases in stormwater water runoff during precipitation events.

 doi:10.20944/preprints202211.0519.v1

Both peak flows and the total volume of stormwater runoff increase because of the increase in impervious surfaces. It is also expected that climate change will affect the intensity and accumulation of rainfall, further complicating the ability to design resilient, adaptable, and long-lasting Stormwater Control Measures (SCMs) [1–5]. SCMs are implemented to improve water quality and manage the increased quantity of stormwater runoff by restoring the hydrologic cycle disrupted by development [6–9]. The past performance of an SCM can be used to forecast future behavior; however, further changes might need to be implemented to consider the effects of climate change [10–12].

Some of the most frequently used SCMs are green roofs, retention/detention basins, bio-retention systems, bioswales, rain gardens, and pervious pavement systems. Retention basins typically capture runoff and maintain a permanent body of water. Detention basins, which can assist with flood control and peak flow reduction, capture water before releasing it downstream. Both retention and detention basins improve water quality primarily by slowing the flow of water enough to allow sediments to fall out of suspension [13,14]. Often contaminants, such as nitrogen and phosphorus, are adsorbed onto sediments [15–17]. Basins can be drained by several mechanisms: continuous slow release by restricting the size of the outflow orifice, manually manipulating a release gate, or by remotely and dynamically controlling the outflow gate, e.g., real-time controls (RTC) [18–22]. A fixed outlet opening is the least expensive option, but the flowrate is directly related to the available water depth in the basin and cannot be controlled [23,24]. Manually manipulating a gate to release water is not practical for most systems because of the high cost and availability of labor. RTC allow for the gate to be manipulated automatically to release the stored water based upon certain triggers to reduce peak flow intensity and increasing the volume of water that can be detained [25]. In RTC systems, the timing of the opening and closing of the outlet gate is controlled by a computer program. This program keeps the gate closed to allow the basin to fill up to the desired water depth, the gate is then programmed to open and close to regulate the release of water from the basin [26–29]. The goals of this RTC system were to:

1. Maximize the volume of stormwater captured during storm events by ensuring that space is available in the basin by releasing water retained by the basin in advance of an upcoming storm.
2. Increase the residence time of stormwater in the basin, to allow sediment to settle, with a maximum retention time of two weeks to provide a healthy environment for the basin's flora and fauna.
3. Reduce erosion downstream by reducing the peak flowrate.
4. Reduce or eliminate the number of overtopping events.

The first and second goals reveal a tension between increasing the residence time to enhance water quality through the settlement of sediment and the pollutants attached to them and allowing the water to drain to prepare the basin for an upcoming rain event and to avoid negative impacts on vegetation types that are not conducive to continuous saturation [30–35]. Hence, the program for the RTC system adheres to multiple controlling rules to optimize the performance of the SCM to meet water quantity and quality goals.

The purpose of this study was to optimize the RTC performance of the outlet structure using machine learning. The machine learning approach consisted of data preparation, model training, model optimization and model comparison. Open-source Python libraries were used to facilitate this process and build the machine learning models. The initial model that was trained on historical data was a multiple linear regression and the second was a long short-term memory network. Both multiple linear regression and long short-term memory have been used by previous researchers as data driven models to predict streamflow, water table depth, and urban flooding; however, there is a need for further investigation on how these techniques can be used to optimize RTC systems for an individual SCM using rainfall data, which is commonly available [36–43]. An exploratory

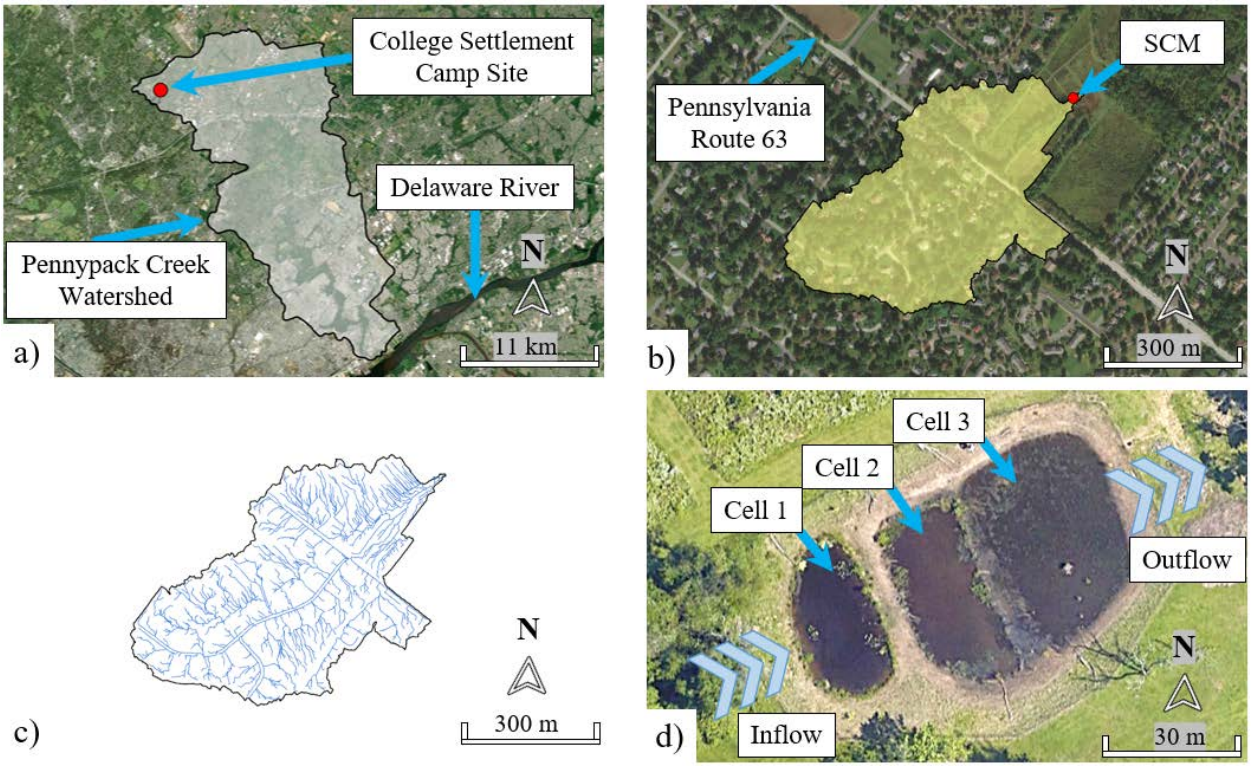
Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 29 November 2022 doi:10.20944/preprints202211.0519.v1

data analysis approach was used to analyze the four years of historical data at the research site to develop a program to optimize basin performance.

2. Site Description

A vegetated multi-stage basin located at the headwaters of the Pennypack Creek on the College Settlement Camp in Horsham, PA is the focus of this study. The Pennypack originates at the location of the SCM and flows roughly 24 km southeastward to its junction with the Delaware River in Philadelphia (Fig. 1a) [44, 45]. This SCM was designed to manage a 51-millimeter storm from a 0.22 km^2 drainage area, which is a mix of residential and open field land (Fig. 1.b). The contributing watershed is 24% impervious with a 1.3% slope. The residential area was built before stormwater controls were required, and this uncontrolled stormwater water flows from the residential area to a swale that leads to the SCM (Fig. 1.c) [46]. In addition to the runoff from the residential area, water from the open field flows overland directly into the SCM, so there is no single point of entry.

The SCM has three cells: two retention basins followed by one detention basin with an overall surface area of 2860 m^2 (Fig. 1.d). Stormwater enters the system and fills up the first cell which functions as a sedimentation basin. If there is sufficient volume, water then overflows to the second cell over a 0.61 m high berm. Likewise, if there is sufficient volume, the water then overflows into the third basin over a 0.23 m berm. During larger storms all three basins are filled with water and the berms are submerged. The water moves through all three cells before leaving the SCM via an outlet structure. Final outflow from the system occurs through a pipe at the end of cell three. This pipe is hydraulically connected to the outlet gate which was initially outfitted with a manually controlled system [47]. The outlet structure was retrofitted in April 2021 with an automated gate. This retrofit allowed the gate to be programmed and remotely controlled to manage the water level inside the last cell (cell three) by commanding the gate actuator to open or close the gate. Both the manual and automated gates were manufactured by Agri-drain. The cells were planted with native plants that could withstand both wet and dry conditions.



 doi:10.20944/preprints202211.0519.v1

Fig. 1. Study area, a) Pennypack Creek Watershed, b) Stormwater Control measures at College Settlement Camp, Horsham, Pennsylvania, c) Streamlines within the SCM drainage area, d) The multi-stage basin shortly after construction before vegetation was established.

3. Methodology

This study investigated the performance improvement of an SCM by implementing an RTC trained on historical system behavior data observed between June 2017 and June 2021. This period was selected based on the installation and calibration of the instrumentation and availability of accurate data. The data were collected in 5-minute intervals. Machine learning was used to predict the water level in cell three to improve the RTC performance (Fig. 2). The initial prediction approach focused on predicting the water level with an algorithm that ingested every precipitation event along with the associated cell measurements to train the model. Multiple linear regression and long short-term memory model performance was studied and compared. Data collection, data preprocessing, exploratory data analysis, feature engineering, model training, and model evaluations are discussed in the following sections. Model deployment considerations related to RTC and SCM performance are provided in the Discussion.

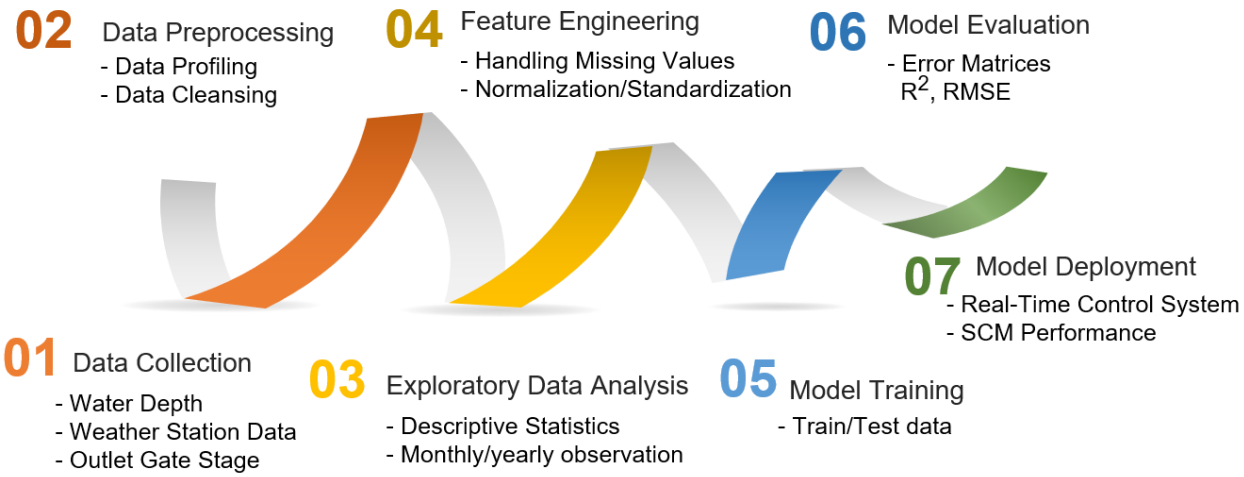


Fig. 2. The methodology pipeline investigating the performance improvement of a low-cost real-time stormwater control system.

3.1. Data Collection

Step 1: A machine learning approach was initiated using data from onsite sensors within the SCM (OTT Hydromet Compact Bubbler Sensors for water level in each cell) and the onsite weather station (Fig. 3). Downstream of the SCM is an H-flume that is monitored using an OTT Compact Bubbler Sensor for water level. A calibrated flume equation uses water level as in input to produce an outflow at 5-minute intervals. The data collected from these sensors between June 2017 and June 2021 provided attributes (otherwise known as features in data science) for the initial machine learning analysis (Table 1). The weather station was installed in April 2016 to monitor precipitation, temperature, humidity, solar radiation, wind speed, and barometric pressure. Prior to automation in April 2021, a robust set of data for each gate condition was obtained from the gate stage changes. Manipulating the gate changes the water level in cell three because there is a berm separating cell two from cell three and cell two from cell one (Fig. 4).

 doi:10.20944/preprints202211.0519.v1

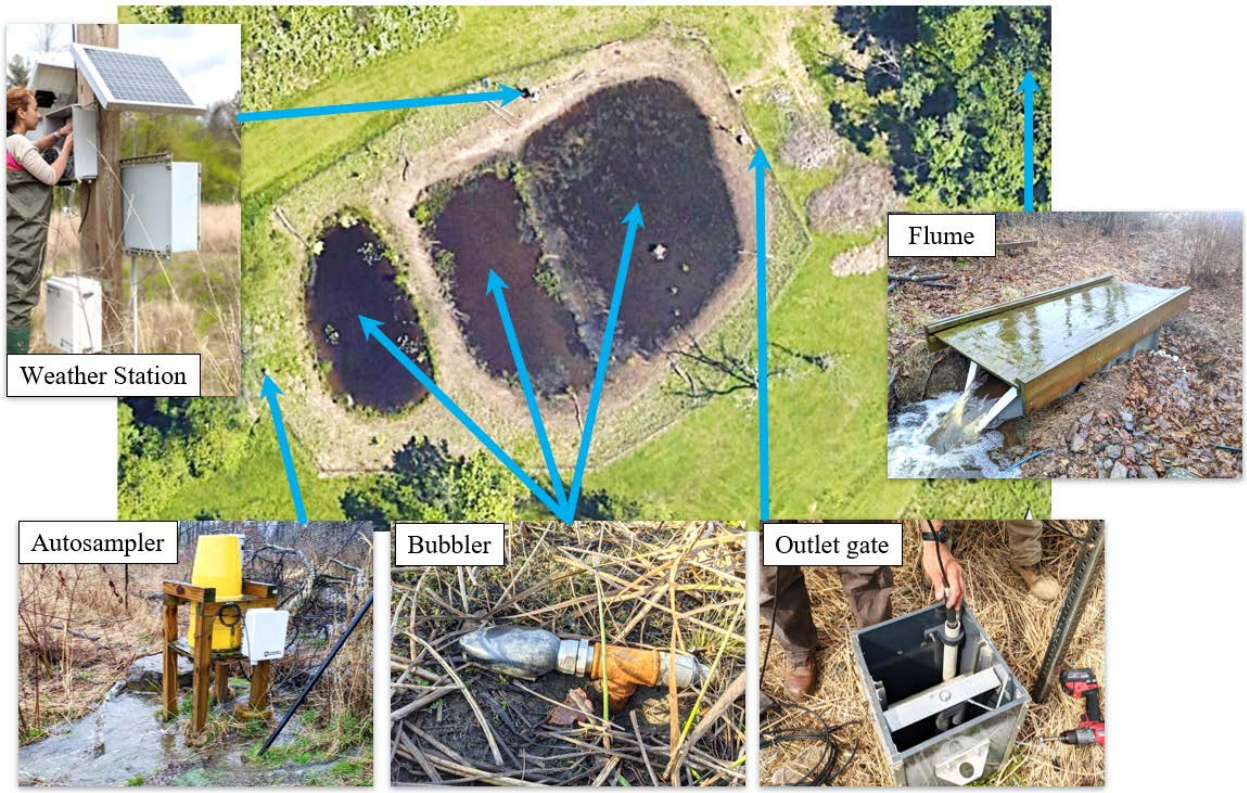


Fig. 3. College Settlement SCM instrumentation and their location.

Table 1. List of the variables used for exploratory data and predictive analysis

Predictor variables	**Unit**	**Descriptions**
Date	Time	The date and time in five minutes intervals
Precipitation	mm	Accumulated rainfall at site's weather station next to SCM
Temperature	Fahrenheit	Atmospheric temperature of the weather at site's weather station
Humidity	Percent	Relative humidity at site's weather station
BaroPress	Inches of Hg	Barometric pressure at site's weather station
C1Level	Meters	Water level in cell one
C2Level	Meters	Water level in cell two
FlumeLevel	Meters	Water level in the outlet flume
Gate stage	Categorical/ Nominal	The gate stages of being opened or closed
Target variable	**Unit**	**Description**
C3Level	Meters	Water Level in cell three

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 29 November 2022 doi:10.20944/preprints202211.0519.v1

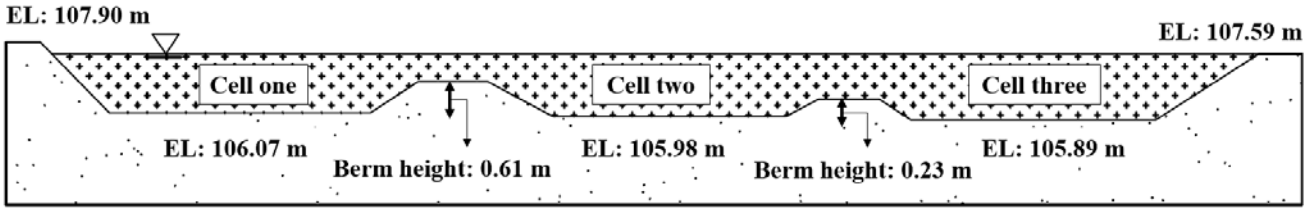


Fig. 4. A cross section of the multi-stage basin.

3.2. Data Preprocessing

Step 2: Data preprocessing included examining and cleansing the data based on available trusted information for each variable. Data examination was initiated by plotting the variables and calculating statistical summaries for each variable. Temperatures below freezing and observed clogging at the end of the flume bubbler hose produced erroneous data that needed to be addressed during the cleansing process. Flume water depth and outflow gate values were discarded when the temperature was below freezing since sensor malfunctions resulted in null values in all the observations. Detecting inaccurately reported values helped improve model efficiency. The dataset was also revised based on the upper and lower detection limits for sensors and maximum site capacity. As the result, incorrect data was converted to null values. After data cleansing, the remaining outliers accurately reflected the water level since they were directly related to storm conditions; these data were critical to the dataset used in the prediction models and could not be removed.

3.3. Exploratory Data Analysis

Step 3: Exploratory data analysis is the process of performing initial investigations on data to discover attributes and characteristics, identify anomalies, and check assumptions using summary statistics and visualizations. The dataset consisted of two groups: weather components and water quantities. Accumulated observed rainfall, temperature, relative humidity, and barometric pressure were part of the weather component. The water level inside the three cells and the flume, as well as the gate stage fell into the category of water quantity. To understand the hidden pattern of the variables' distributions, an initial investigation was completed focusing on descriptive statistics and extreme value detection. Table 2 shows the descriptive statistics of all the variables and the values checked with the possible data range related to this site to characterize the distribution of features. Fig. 6 illustrates the water depth box plots for all four locations (three cells and the flume) grouped by gate stage. Over the four years of data, the gate had three stages: opened, closed, and partially opened, where the partially opened stage was only an option before RTC implementation. After April 2021 the gate was either fully opened or fully closed. Each of the outliers are represented by a black point. Each single extreme value represents a storm event that caused the increase in water depth.

Table 2. Descriptive Statistics of the variables

Variable	Count	Mean	Std	Min	25%	50%	75%	Max
Precipitation	436961	0.01	0.13	0	0	0	0	13.9
Temperature	419601	53.33	19.73	-5.3	37.8	54.1	69.7	98.4
Humidity	435886	70.00	20.76	0	57	74	88	95
BaroPress	435293	28.87	4.89	0	29.54	29.68	29.83	30.47
C1Level	434831	0.63	0.08	0.5	0.6	0.62	0.64	1.56
C2Level	435925	0.29	0.23	0	0.19	0.23	0.25	1.69

 doi:10.20944/preprints202211.0519.v1

C3Level	435919	0.22	0.33	0	0	0.1	0.24	1.85
FlumeLevel	432438	0.04	0.05	0	0.02	0.03	0.053	0.65

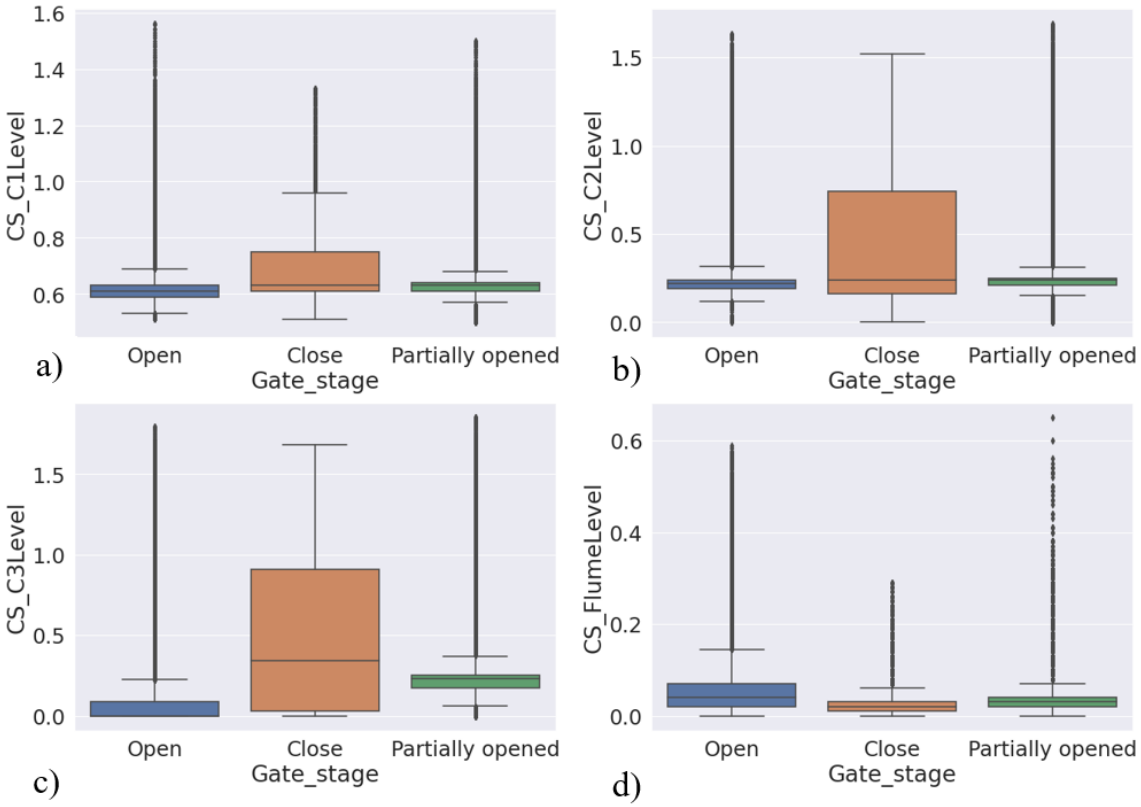


Fig. 5. Boxplot for water depth grouped by gate stage, a) cell one, b) cell two, c) cell three, and d) flume.

An analysis of the water levels in the cells by year and month (Fig. 7) indicated that the program should change monthly to improve performance. In this region, more intense and more frequent rainfall occurs during summer, especially June and July. Thus, in the summer the gate needs to be open for longer and with more frequency in support of the goal of maximizing the volume of stormwater controlled. In advance of an upcoming storm the SCM must be emptied to provide capacity to accept stormwater. The timing and duration of the gate opening and closing is therefore more critical when storm events are larger and more frequent.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 29 November 2022 doi:10.20944/preprints202211.0519.v1

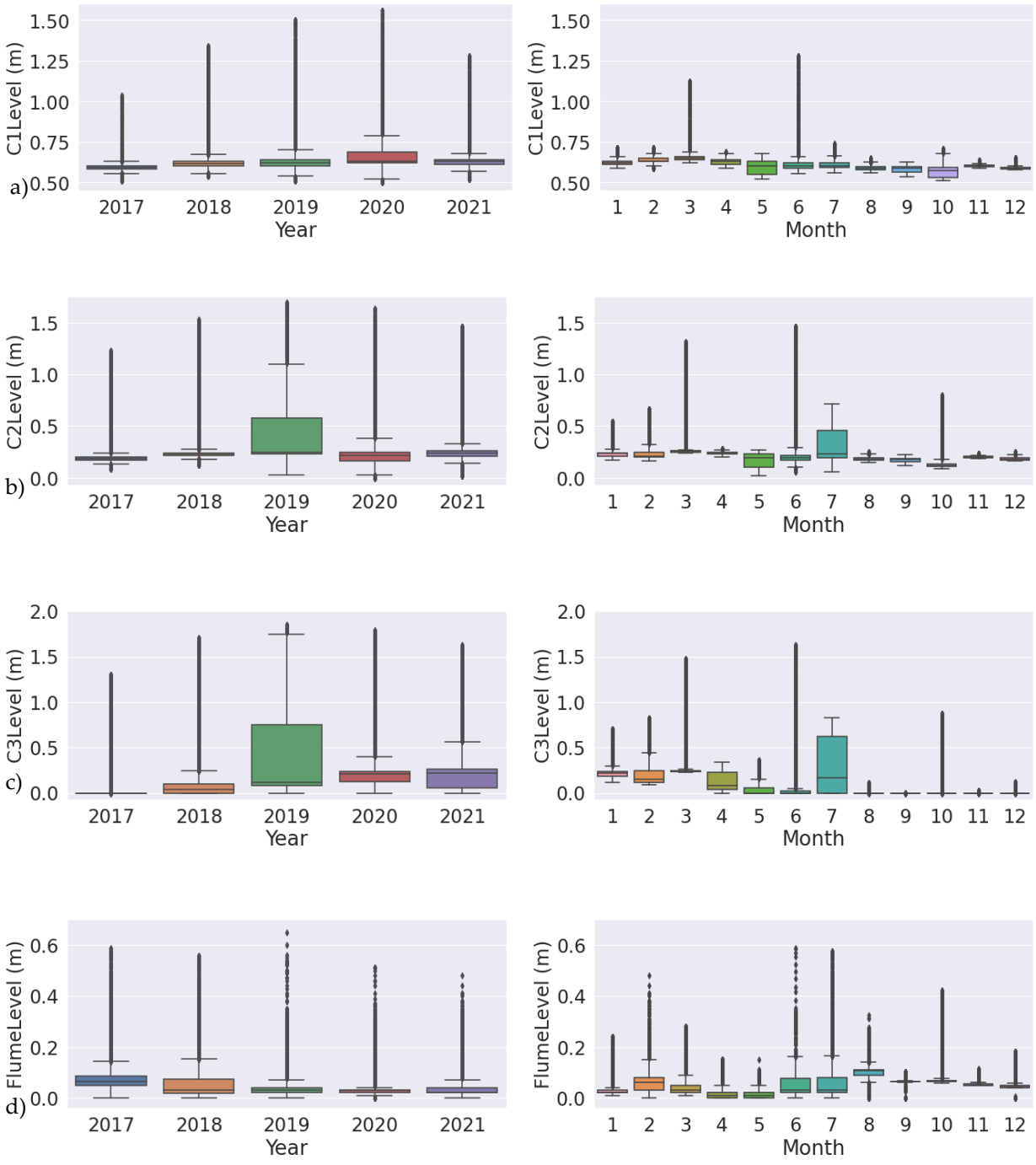


Fig. 6. Yearly and monthly distribution of the water quantity variables using boxplots; a) water level in cell one, b) water level in cell two, c) water level in cell three, and d) water level in flume.

3.4. Feature Engineering

Step 4: Following the exploratory data analysis, feature engineering was conducted on the dataset. This step was required before the training/testing step of the LSTM algorithm as it prepared the dataset for predictive analysis with the best performance and minimum error. This stage included the typical feature engineering processes such handling missing values and data standardization and normalization. The total number of rows for this data set was 436,961 for each of the 5-minute interval variables. Among all the datasets there were 17,360; 1,075; 1,668; 2,130; 1,036; 1,042; and 4,253 undefined (Not a

 doi:10.20944/preprints202211.0519.v1

Number or NaN) values for Temperature, Humidity, BaroPress, C1Level, C2Level, C3Level, and FlumeLevel, respectively. Although imputation of null values by mean or median is a common method for handling missing values, three other methods of dropping, filling (ffill method in Python), and interpolating were used and compared to determine which method was most appropriate. Imputation of the NaN values during a storm event, with either mean or median, was not appropriate for these datasets. Dropping any rows of data with a NaN value, filling NaN values for each period with the last observed value, and linearly interpolating a replacement for the NaN values for each period between the first and last observations were the three methods considered in this study.

Through the data standardization process, the values of a variable were rescaled so that the variable had a mean of 0 and a variance of 1 (or **Z-score normalization),** which is identical to the bell-shaped normal distribution curve. Normalization was an important step for training and testing the neural network algorithm. The long short-term memory, recurrent neural network model used the gradient descent technique where feature values affect the step-size of each iteration. Smooth progress towards finding the global minima in gradient descent required the update of the steps at the same rate for all the feature values. Standardized variables are a prerequisite of reaching the minima in the gradient descent process. All the values in the water depth series were normalized to prepare the training dataset for the long short-term memory model. Equation 1 shows the normalization formula.

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (1)$$

The difference between the water depth value and the minimum of the entire water depth series was divided by the range of the series and provided the standardized data which was used in the training and testing process of the LSTM. The entire normalized water depth series was split into two portions i.e., a training set that was used to train the model and a testing set that was used to test and evaluate the model. Seventy percent of the dataset was used for training and 30% was used for testing.

3.5. Model Training

Step 5: After feature engineering and data normalizations, splitting the dataset into training and testing sets was the next step. LabelEncoder is a normalization method in Python that converts non-numerical labels (categorical values) to numerical labels so they can be analyzed through machine learning algorithms. The "Gate Stage" variable, which had a categorial feature with the values of open, close, and partially opened, was transformed using the LabelEncoder method to a numerical feature with the values of 0, 1, and 2 for the analysis.

3.5.1. Multiple Linear Regression

There was a total of nine variables (Table 1) used in the multiple linear regression with the target variable being the water level in cell three (which is the cell adjacent to the RTC system at the outlet structure). The multiple linear regression algorithm was imported from the "sklearn" library in Python and used to perform training. Fig. 8 shows the comparison between predictions and the original (observation) target variable. As one of the ordinary least squares regressors, linear regression was used to fit a linear model to all the features with the coefficient (β), where the coefficients were not raised to any power and did not combine in any term to minimize the residual sum of squares between the observed and predicted water depths [48] (Eq. 2).

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n \quad (2)$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 29 November 2022 doi:10.20944/preprints202211.0519.v1

All three methods for handling missing values used both Ridge and Lasso (least absolute shrinkage and selection operator) regressions. The difference between these methods is what is dubbed the penalty, or regularization, term. Ridge regression employs L2 regularization to penalize the magnitude of the coefficients, which alleviates some of the problems associated with ordinary least squares [49]. Lasso regression employs L1 regularization and panelized terms based on the sum of the coefficient absolute values [50]. For the Ridge regression, the L2 regularization was implemented by imposing a penalty equal to the square of the coefficients' magnitude and minimizing the sum of coefficients' square (Eq.3). Lasso regression employs L1 regularization by considering an absolute value of the coefficients and minimizing the sum of coefficients' absolute value (Eq.4). The alpha (α) coefficient helped the minimization of these two previous objectives by multiplying the alpha value by the summation term and controlling the penalty weight represented in Equations 3 and 4 for both regularized methods.

$$\text{Penalty term: } L2 = \alpha \sum_{i=1}^{n} {\beta_i}^2 \quad (3)$$

$$\text{Penalty term: } L1 = \alpha \sum_{i=1}^{n} |\beta_i| \quad (4)$$

Alpha is a penalty term which indicates how much constraint will be applied to the equation. Thus, when the alpha is set to zero, the equation transforms to the linear regression model, while a higher value of alpha penalizes the optimization function. The best fitted alpha within the range of 10^{-7} to 10 for both Ridge and Lasso was found to be 0.001.

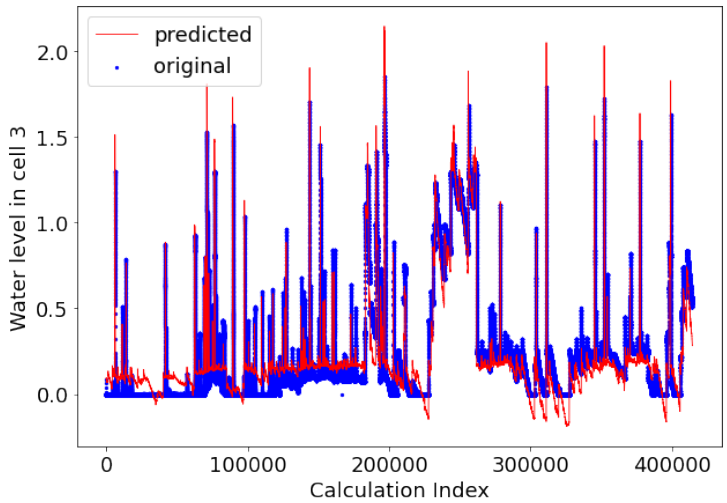


Fig. 8. The comparison between predictions and original target values by MLR.

3.5.2. Long Short-Term Memory

Long short-term memory is a type of recurrent neural network frequently used for time series forecasting and is often used when variables are dependent on the previous data in the series [51,52]. Long short-term memory has the ability to capture the long-term dependencies among the predictor and target variables [53,54]. Long short-term memory feedback connections are the principal component of processing and recalling long-term information; this is a unique feature which differentiates it from a traditional multilayer perceptron method. The multilayer perceptron method is a type of artificial neural network that uses a feed forward method for the prediction process with three main layers as the input layer, hidden layer, and the output layer [55,56].). This unique feature of the long short-term memory approach is utilized in processing the time series, e.g., all the data points for the water level in cell three were treated independently while considering their relative timing to each other.

In the long short-term memory model both long-term (c[t-1]) and short-term memory (h[t-1]) are processed through multiple gates to filter the data flow. Three gates, the forget

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 29 November 2022 doi:10.20944/preprints202211.0519.v1

gate (f_g) (Eq. 5), the input gate (i_g) (Eq. 6), and the output gate (o_g) (Eq. 7), control the data processing by writing, discarding, and reading each data point respectively (Fig. 9).

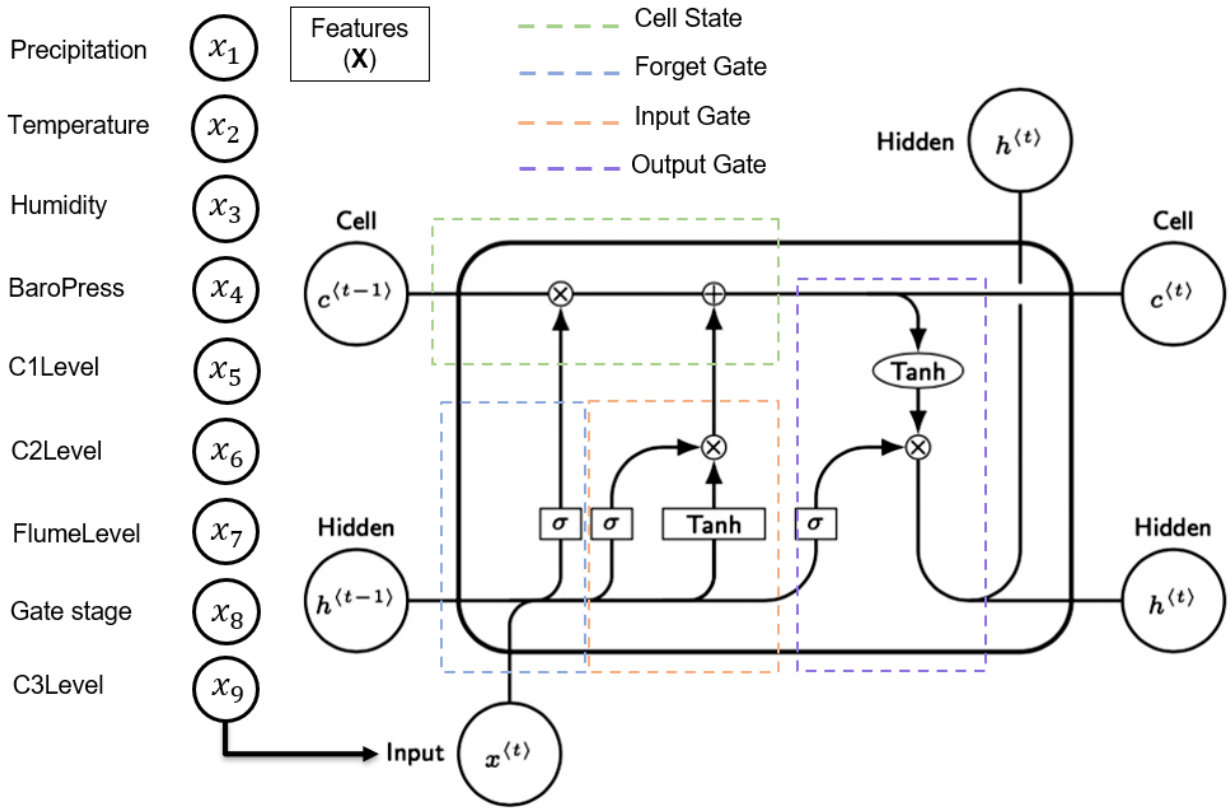


Fig. 7. Schematic representation of a long short-term memory structure with its four different gate withing each cell.

Long-term data were injected and passed through a filtration process in the forget gate where the unnecessary information was rejected. Based on the sigmoid activation filtration, the forget gate filtered out irrelevant data. The range of the activation function was 0 and 1 showing the gate options as opened or closed and quantifying the importance of new data entering the cell or not. The input gate regulated the flow of both short-term and long-term information by filtering out information using binary activation functions the same way as the forget gate. The information from prior inputs was used by the output gates to adjust the value of the following hidden state. Based on an understanding of recent inputs, the output gates regulated the value of the next hidden state. All cell operations are presented in the following equations.

$$f_g = sigmoid(X_t V_f + h_{t-1} W_f + b_f) \quad (5)$$

$$i_g = sigmoid(X_t V_i + h_{t-1} W_i + b_i) \quad (6)$$

$$o_g = sigmoid(X_t V_o + h_{t-1} W_o + b_o) \quad (7)$$

$$h_t = o_g \odot \tanh(C_t) \quad (8)$$

$$C_t = i_g \odot \tilde{C}_t + f_g \odot C_{t-1} \quad (9)$$

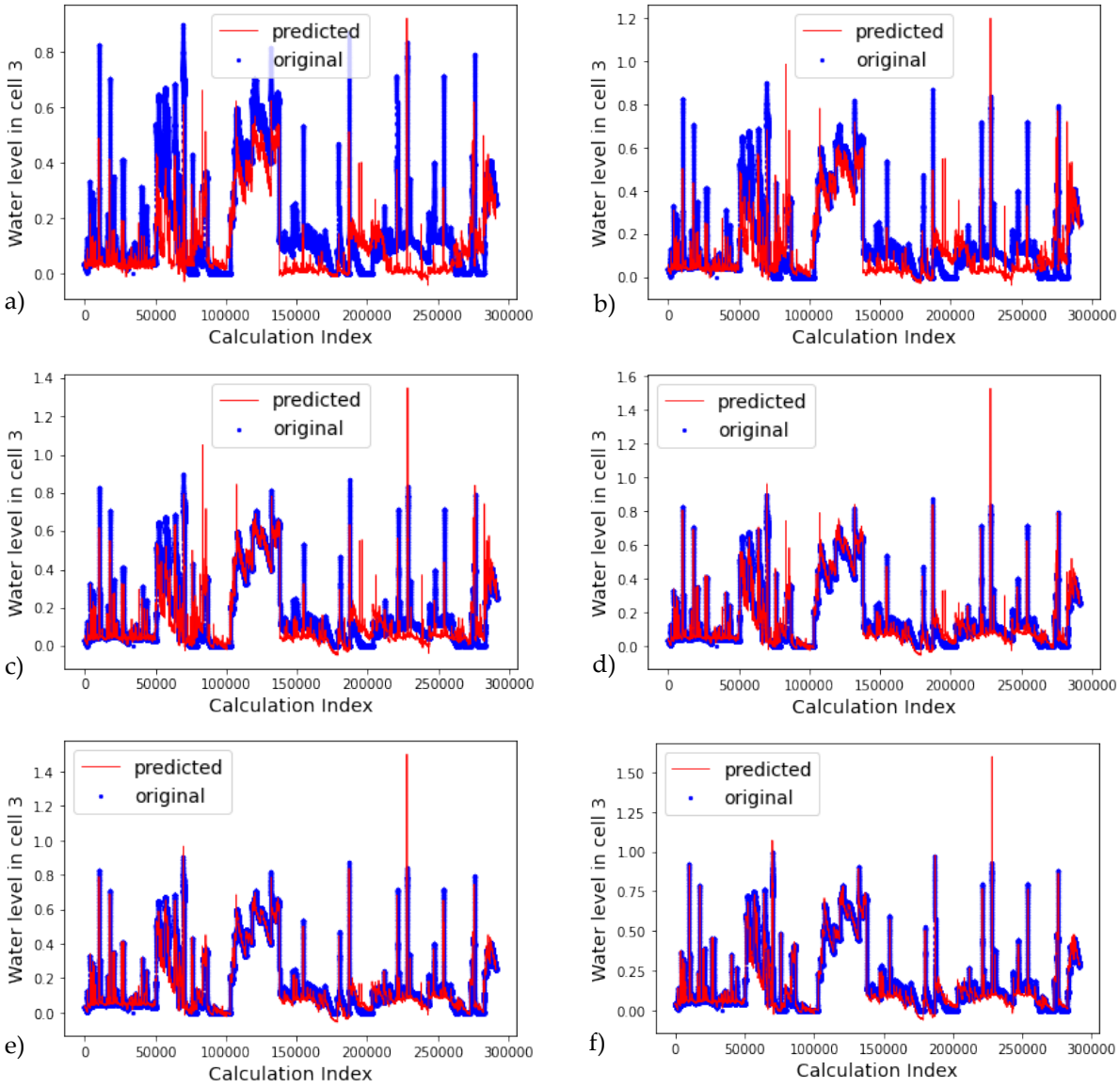
$$\tilde{C}_t = \tanh(X_t V_c + h_{t-1} W_c + b_c) \quad (10)$$

The Hadamard product is indicated by the operator $\odot$ (element-wise multiplication). The hidden state is connected to the short-term memory by a vector called h_t (Eq. 8). The cell state is represented by Ct (Eq. 9) and linked to the long-term memory. $\tilde{C}_t$, which is the candidate for the cell state at time lag t, filters and stores effective and crucial data (Eq. 10). The input gate, forget gate, output gate, and cell state all utilize use of various weight

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 29 November 2022 doi:10.20944/preprints202211.0519.v1

matrices. Hence, the long short-term memory model implemented the prefixes W_i, W_f, W_o, W_c , V_i, V_f, V_o, V_c, and b_i, b_f, b_o, b_c as biases and weight matrices through the overall process for the current input, X_t, prediction.

Fig. 10 compares the predictions and the original water level in the last cell by the long short-term memory model with different lead times ranging from 5 minutes to 12 hours. As it was expected the smaller the lead time chosen, the more precise the prediction based on the previous specified step by the long short-term memory model. The larger values for lead time (12 hours) resulted in more error and lower accuracy of the next step prediction as opposed to the smallest lead time (5 minutes).



Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 29 November 2022 doi:10.20944/preprints202211.0519.v1

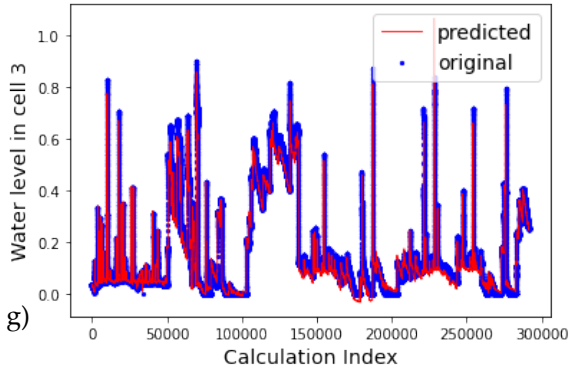
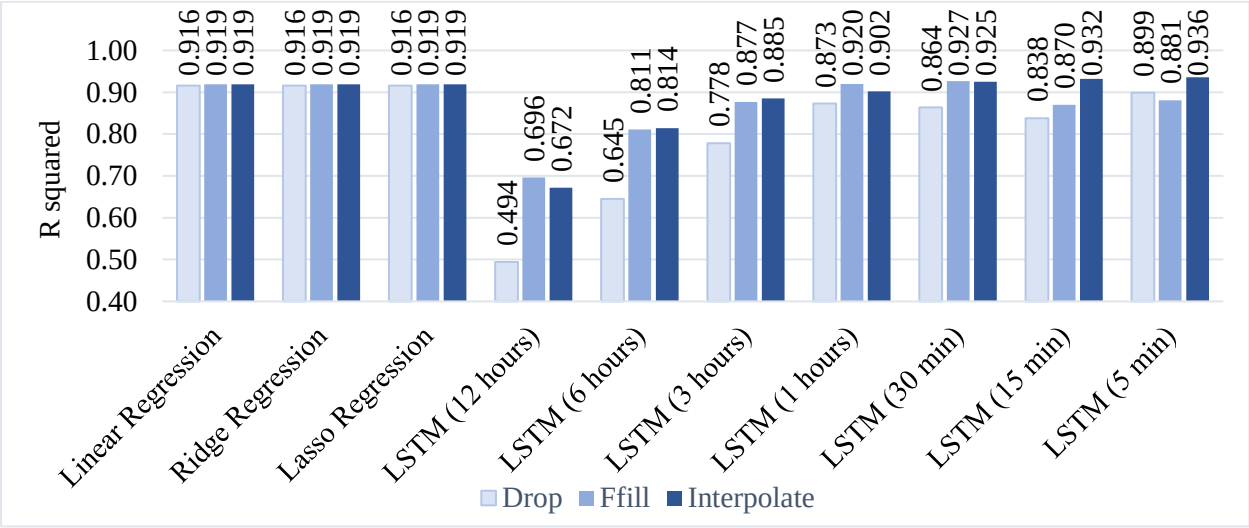


Fig. 8. The comparison between predictions and original target values by the long short-term memory model with different lead times; a) 12 hours, b) 6 hours, c) 3 hours, d) 1 hour, e) 30 minutes, f) 15 minutes, and g) 5 minutes.

3.6. Model Evaluation

Step 6: Model evaluation consisted of two standard error metrics, R-squared (R^2) and root mean square error (RMSE), to measure the goodness-of-fit of the regression analysis [55]. Since the squared term magnifies larger errors more than smaller ones, the RMSE is more sensitive to major errors [57,58]. The water depth in cell three, which was the target variable, did not deviate significantly from the average value most of the time during each year. Amplifying the changes in water depth for storm event conditions played an important role in determining the functioning requirement of the outlet structure. Rather than comparing predictions and observations, these error metrics provided a quantitative comparison between different models. The performance of the multiple linear regression and the long short-term memory model was improved by reducing the time lead. Fig. 11 illustrates both R-squared and RMSE results for all 30 models. The best predictive accuracy was associated with the lowest RMSE score and R-squared values closest to 1. The result of the model evaluation was the selection of the best fitted model.



Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 29 November 2022 doi:10.20944/preprints202211.0519.v1

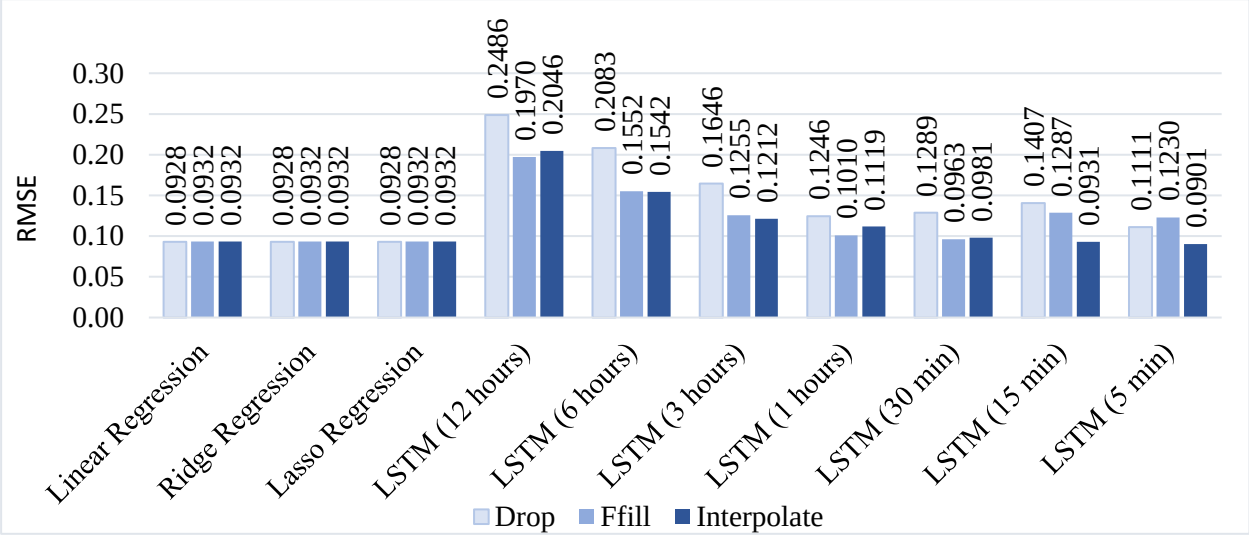


Fig. 9. Model performance comparison by a) R-squared, and b) RMSE scores for multiple linear regression (linear, Ridge, and Lasso) and long short-term memory (LSTM) methods.

4. Results and Discussion

The target variable (water depth in cell three) was predicted using multiple linear regression and the long short-term memory methods. Different lead times, ranging from 5 minutes to 12 hours, were selected and the long short-term memory model was trained for each interval and for each method of handling missing values. A total of 30 models were evaluated including three multiple linear regression methods, having Ridge and Lasso and seven the long short-term memory methods with various lead times (Fig. 11). Each of these 10 models were implemented and trained with three methods for handling missing values. Predicted values were compared to the observed values with two error metrics. The overall R-squared and RMSE for linear regressions were 0.92 and 0.093, respectively, but the long short-term memory model produced better fitted models with higher accuracy and lower errors for all the selected lead times, ranging from 5 minutes to 12 hours.

Although, dropping NaN values produced more accurate predictions (R squared) and less error (RMSE) for most of the models, this method resulted in discarding 5% (22,564) rows of the data. Furthermore, some of the data discarded were from intense storm events, which need to be included to develop a model that serves the goal of improving the performance of the RTC program for a range of storm sizes. This SCM was designed to capture a 51-millimeter storm and any accumulated rainfall below the design capacity was considered as "typical" storm event for this site, and any storm with more rainfall was considered an intense storm event. Thus, the method of dropping values was not used because too many values were excluded. Instead, the NaN values were replaced by interpolated values. This method resulted in more accurate predictions in overall.

Once the long short-term memory model and the method of handling the NaN values was determined, the effect of changing the lead time was the evaluated. Seven different lead times, ranging from 5 minutes to 12 hours were considered (Fig. 12). Smaller lead times produced more accurate results as reflected by the R-squared and RMSE values. However, the improvement in accuracy after 30 minutes is minimal, e.g., the R-squared values decreased by 0.2%, and the RMSE increased by 2.4%, from 5 minutes to 30 minutes. In conclusion, using a 30-minute lead time saves significant computational effort with a minimal decrease in accuracy. Practically, this lead time also allows for the gate to respond to changing weather conditions.

 doi:10.20944/preprints202211.0519.v1

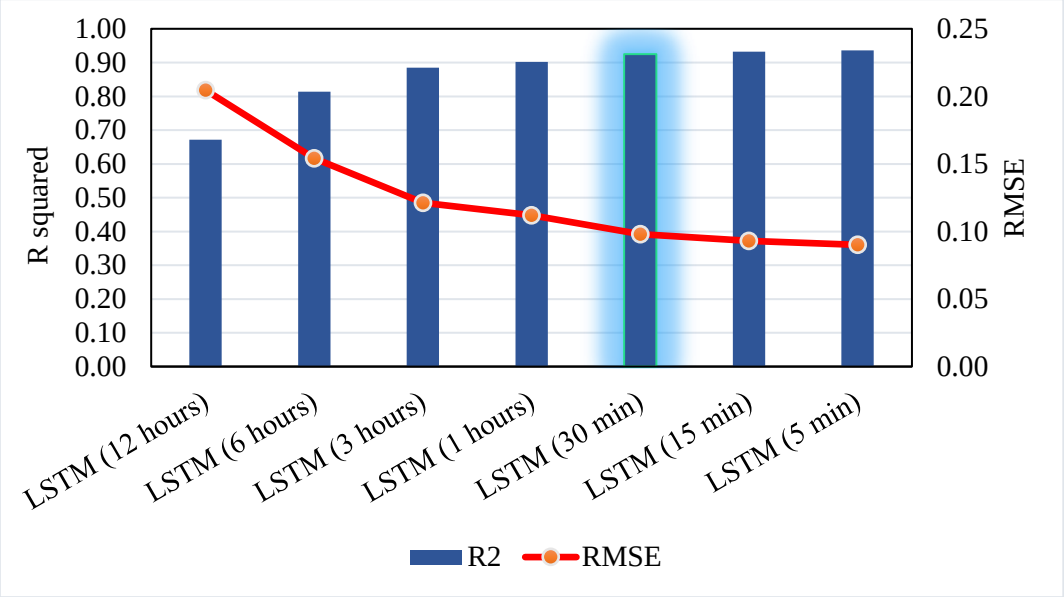


Fig. 10. Comparison of the long short-term memory (LSTM) models with varying lead times with interpolation method for NaN values.

5. Conclusion

Machine learning was effectively used to improve programming for a low-cost RTC multistage SCM basin in southeastern Pennsylvania. Multiple linear regression and the long short-term memory models were compared, and the long short-term memory model was deemed to be superior because of the higher accuracy and lower error of the analyzed models. Once the long short-term memory model was selected, several methods were considered for removing NaN values. Linear interpolation was selected over dropping and filling as this method included the largest amount of data, including intense storm events. Lastly, different lead times were considered. A lead time of 30 minutes yielded accurate results with acceptable computational effort. The selected model will control the gate to meet the four goals of the RTC installation which were to increase the volume controlled, decrease peak flows, minimize downstream erosion, and optimize residence time to balance pollutant removal and time of inundation.

Prior to the installation of the RTC, the maximum volume controlled when the gate was opened was the volume of the first two cells since water could freely exit the third cell. After the installation of the RTC, the gate could be closed in advance of or during storm events which increased the controlled volume by retaining water in the third cell. In addition, during larger storm events, water is able to rise above the berms because they are below the surface elevation. In this case, the cells are hydraulically connected and the SCM functions as one large basin (Fig. 4). Hence, the controlled volume increased from approximately 495 m^3 to 1600 m^3 by strategically closing the gate.

The SCM was designed to manage a 50.8-millimeter storm, thus all runoff generated from smaller storms were controlled by the SCM when the gated was closed, thus decreasing peak flow during storms, which in turn minimized downstream erosion. The residence time for the captured runoff in the third cell increased from six hours to two weeks after the RTC was implemented.

The program releases excess runoff to prevent overtopping once there is more than 50.8 mm of accumulated rainfall. Since the RTC was deployed, there were two instances of overtopping. Hurricane Ida (September 1st, 2021) was an extreme event with over 203 mm of accumulated rainfall. It is unlikely that overtopping during such an extreme event can be fully prevented. The second overtopping event occurred on July 7th, 2022, as a result of a storm with 82.2 mm of rainfall. During this event the gate was not opened soon

enough to provide the required capacity for the incoming runoff during the storm. That event revealed the need for further examination of the programming of the gate to respond to events of this magnitude. Since that event, the SCM was not experienced an event of this size, so the effectiveness of the changes has not been tested.

As more data are collected the model will continue to be trained. In the future, the opening and closing of the gate can be predicted in advance based on the water depth prediction due to the rainfall forecast. The RTC performance of the automatic outlet structure can be improved by the long short-term memory model prediction. Up to this point, the optimization process has focused on the crucial conditions that occur when rainfall intensity is extremely high. The incorporation of storm intensity will allow for more realistic predictions by combining consecutive precipitation events into more significant storm events. Long short-term memory model predictions allow the outlet structure to systematically control the gate to be prepared for the next storm event to provide the maximum volume capacity to capture incoming stormwater runoff.

Successful prediction models based on the time-series dataset and RTC performance improvement will provide the opportunity to expand the use of this technology. In addition, further studies will be performed to determine if the model can effectively control the gate with fewer inputs. Retrofitting existing statically controlled SCMs with dynamic controls will improve the resiliency and adaptability of these SCMs.

Acknowledgements

This study was supported by the William Penn Foundation as part of the Delaware River Watershed Initiative and the partners of the Villanova Center of Resilient Water System (VCRWS) of Villanova University. The findings represent those of the authors and not the funding agency.

Competing interests

The authors declare no competing interests.

Code and data availability.

The code and data to reproduce our results is available at https://github.com/MShivaKh/Khosravi-CS_LSTM, last access: 31 October 2022.

Funding

William Penn foundation and Villanova Center of Resilient Water System.

Corresponding author

Correspondence and requests for materials should be addressed to Marzieh Khosravi

References

1. Kerkez, B.; Gruden, C.; Lewis, M.; Montestruque, L.; Quigley, M.; Wong, B.; Bedig, A.; Kertesz, R.; Braun, T.; Cadwalader, O.; et al. Smarter Stormwater Systems. *Environ. Sci. Technol.* **2016**, *50*, 7267–7273, doi:10.1021/acs.est.5b05870.
2. Ghaith, M.; Siam, A.; Li, Z.; El-Dakhakhni, W. Hybrid Hydrological Data-Driven Approach for Daily Streamflow Forecasting. *J. Hydrol. Eng.* **2020**, *25*, 04019063, doi:10.1061/(ASCE)HE.1943-5584.0001866.
3. Khosravi, M.; Arellano, D. Selection of Adequate EPS-Block Geofoam for Use in Embankments Subjected to Seismic Loads. *Preprints*, **2022**, 2022100074, doi: 10.20944/preprints202210.0074.v1.
4. Halder, S.; Saha, U. Future Projection of Extreme Rainfall for Flood Management Due to Climate Change in an Urban Area. *J. Sustain. Water Built Environ.* **2021**, *7*, 04021012, doi:10.1061/JSWBAY.0000954.
5. Martel, J.-L.; Brissette, F.P.; Lucas-Picher, P.; Troin, M.; Arsenault, R. Climate Change and Rainfall Intensity–Duration–Frequency Curves: Overview of Science and Guidelines for Adaptation. *J. Hydrol. Eng.* **2021**, *26*, 03121001, doi:10.1061/(ASCE)HE.1943-5584.0002122.
6. Gilliom, R.L.; Bell, C.D.; Hogue, T.S.; McCray, J.E. Adequacy of Linear Models for Estimating Stormwater Best Management Practice Treatment Performance. *J. Sustain. Water Built Environ.* **2020**, *6*, 04020016, doi:10.1061/JSWBAY.0000921.
7. Kabbes, K.; Reichenberger, J.; Briggs, C.; Davidson, C.; Perks, A. Water Resources: Sustaining Quality and Quantity. **2017**, 237–253, doi:10.1061/9780784414811.ch16.
8. Coffman, L.S.; Goo, R.; Frederick, R. Low-Impact Development: An Innovative Alternative Approach to Stormwater Management. **2012**, 1–10, doi:10.1061/40430(1999)118.

9. Cheng, M.; Fang, F.; Kinouchi, T.; Navon, I.M.; Pain, C.C. Long Lead-Time Daily and Monthly Streamflow Forecasting Using Machine Learning Methods. *J. Hydrol.* **2020**, *590*, 125376, doi:10.1016/j.jhydrol.2020.125376.
10. Fathian, F.; Vaheddoost, B. Modeling the Volatility Changes in Lake Urmia Water Level Time Series. *Theor. Appl. Climatol.* **2021**, *143*, 61–72, doi:10.1007/s00704-020-03417-8.
11. Shafizadeh-Moghadam, H.; Valavi, R.; Shahabi, H.; Chapi, K.; Shirzadi, A. Novel Forecasting Approaches Using Combination of Machine Learning and Statistical Models for Flood Susceptibility Mapping. *J. Environ. Manage.* **2018**, *217*, 1–11, doi:10.1016/j.jenvman.2018.03.089.
12. Moura, N. c. b.; Pellegrino, P. r. m.; Martins, J. r. s. Best Management Practices as an Alternative for Flood and Urban Storm Water Control in a Changing Climate. *J. Flood Risk Manag.* **2016**, *9*, 243–254, doi:10.1111/jfr3.12194.
13. Bilodeau, K.; Pelletier, G.; Duchesne, S. Real-Time Control of Stormwater Detention Basins as an Adaptation Measure in Mid-Size Cities. *Urban Water J.* **2018**, *15*, 858–867, doi:10.1080/1573062X.2019.1574844.
14. Muschalla, D.; Vallet, B.; Anctil, F.; Lessard, P.; Pelletier, G.; Vanrolleghem, P.A. Ecohydraulic-Driven Real-Time Control of Stormwater Basins. *J. Hydrol.* **2014**, *511*, 82–91, doi:10.1016/j.jhydrol.2014.01.002.
15. Carpenter, J.F.; Vallet, B.; Pelletier, G.; Lessard, P.; Vanrolleghem, P.A. Pollutant Removal Efficiency of a Retrofitted Stormwater Detention Pond. *Water Qual. Res. J.* **2013**, *49*, 124–134, doi:10.2166/wqrjc.2013.020.
16. Flanagan, K.; Blecken, G.-T.; Österlund, H.; Nordqvist, K.; Viklander, M. Contamination of Urban Stormwater Pond Sediments: A Study of 259 Legacy and Contemporary Organic Substances. *Environ. Sci. Technol.* **2021**, *55*, 3009–3020, doi:10.1021/acs.est.0c07782.
17. Nayeb Yazdi, M.; Scott, D.; Sample, D.J.; Wang, X. Efficacy of a Retention Pond in Treating Stormwater Nutrients and Sediment. *J. Clean. Prod.* **2021**, *290*, 125787, doi:10.1016/j.jclepro.2021.125787.
18. Altami, S.A.; Salman, B. Implementation of IoT-Based Sensor Systems for Smart Stormwater Management. *J. Pipeline Syst. Eng. Pract.* **2022**, *13*, 05022004, doi:10.1061/(ASCE)PS.1949-1204.0000647.
19. Ibrahim, Y.A. Real-Time Control Algorithm for Enhancing Operation of Network of Stormwater Management Facilities. *J. Hydrol. Eng.* **2020**, *25*, 04019065, doi:10.1061/(ASCE)HE.1943-5584.0001881.
20. Hill, D.; Kerkez, B.; Rasekh, A.; Ostfeld, A.; Minsker, B.; Banks, M.K. Sensing and Cyberinfrastructure for Smarter Water Management: The Promise and Challenge of Ubiquity. *J. Water Resour. Plan. Manag.* **2014**, *140*, 01814002, doi:10.1061/(ASCE)WR.1943-5452.0000449.
21. Gaborit, E.; Muschalla, D.; Vallet, B.; Vanrolleghem, P.A.; Anctil, F. Improving the Performance of Stormwater Detention Basins by Real-Time Control Using Rainfall Forecasts. *Urban Water J.* **2013**, *10*, 230–246, doi:10.1080/1573062X.2012.726229.
22. Zimmer, A.; Minsker, B.; Schmidt, A.; Ostfeld, A. Evolutionary Algorithm Memory Enhancement for Real-Time CSO Control. **2012**, 2251–2259, doi:10.1061/41114(371)232.
23. Wong, B.P.; Kerkez, B. Real-Time Control of Urban Headwater Catchments Through Linear Feedback: Performance, Analysis, and Site Selection. *Water Resour. Res.* **2018**, *54*, 7309–7330, doi:10.1029/2018WR022657.
24. Niazi, M.; Nietch, C.; Maghrebi, M.; Jackson, N.; Bennett, B.R.; Tryby, M.; Massoudieh, A. Storm Water Management Model: Performance Review and Gap Analysis. *J. Sustain. Water Built Environ.* **2017**, *3*, 10.1061/jswbay.0000817, doi:10.1061/jswbay.0000817.
25. Mullapudi, A.; Bartos, M.; Wong, B.; Kerkez, B. Shaping Streamflow Using a Real-Time Stormwater Control Network. *Sensors* **2018**, *18*, 2259, doi:10.3390/s18072259.
26. Naughton, J.; Sharior, S.; Parolari, A.; Strifling, D.; McDonald, W. Barriers to Real-Time Control of Stormwater Systems. *J. Sustain. Water Built Environ.* **2021**, *7*, 04021016, doi:10.1061/JSWBAY.0000961.
27. Mullapudi, A.; Wong, B.P.; Kerkez, B. Emerging Investigators Series: Building a Theory for Smart Stormwater Systems. *Environ. Sci. Water Res. Technol.* **2017**, *3*, 66–77, doi:10.1039/C6EW00211K.
28. Farrell, A.; Perdikaris, J.; Scheckenberger, R.B. An Evaluation of Stormwater Management Practices to Provide Flood Protection for Watershed-Based Targets. *J. Water Manag. Model.* **2009**, doi:10.14796/JWMM.R235-06.
29. Emerson, C.H.; Welty, C.; Traver, R.G. Watershed-Scale Evaluation of a System of Storm Water Detention Basins. *J. Hydrol. Eng.* **2005**, *10*, 237–242, doi:10.1061/(ASCE)1084-0699(2005)10:3(237).
30. Xu, W.D.; Burns, M.J.; Cherqui, F.; Fletcher, T.D. Enhancing Stormwater Control Measures Using Real-Time Control Technology: A Review. *Urban Water J.* **2021**, *18*, 101–114, doi:10.1080/1573062X.2020.1857797.
31. Persaud, P.P.; Akin, A.A.; Kerkez, B.; McCarthy, D.T.; Hathaway, J.M. Real Time Control Schemes for Improving Water Quality from Bioretention Cells. *Blue-Green Syst.* **2019**, *1*, 55–71, doi:10.2166/bgs.2019.924.
32. Luthy, R.G.; Sharvelle, S.; Dillon, P. Urban Stormwater to Enhance Water Supply. *Environ. Sci. Technol.* **2019**, *53*, 5534–5542, doi:10.1021/acs.est.8b05913.
33. Abdullah Al Mehedi, M.; Reichert, N.; Molkenthin, F. Sensitivity Analysis of Hyporheic Exchange to Small Scale Changes In Gravel-Sand Flumebed Using A Coupled Groundwater-Surface Water Model. **2020**, 20319, doi:10.5194/egusphere-egu2020-20319.
34. Mehedi, M.A.A.; Yazdan, M.M.S. Automated Particle Tracing & Sensitivity Analysis for Residence Time in a Saturated Subsurface Media. *Liquids* **2022**, *2*, 72–84, doi:10.3390/liquids2030006.
35. Mehedi, M.A.A.; Yazdan, M.M.S.; Ahad, M.T.; Akatu, W.; Kumar, R.; Rahman, A. Quantifying Small-Scale Hyporheic Streamlines and Resident Time under Gravel-Sand Streambed Using a Coupled HEC-RAS and MIN3P Model. *Eng* **2022**, *3*, 276–300, doi:10.3390/eng3020021.

36. Khosravi, M.; Arif, S.B.; Ghaseminejad, A.; Tohidi, H.; Shabanian, H. Performance Evaluation of Machine Learning Regressors for Estimating Real Estate House Prices. *Preprints* **2022**, 2022090341, doi: 10.20944/preprints202209.0341.v1.
37. Yazdan, M.M.S.; Khosravia, M.; Saki, S.; Mehedi, M.A.A. Forecasting Energy Consumption Time Series Using Recurrent Neural Network in Tensorflow. *Preprints* **2022**, 2022090404, doi: 10.20944/preprints202209.0404.v1.
38. JianFeng, Z.; Yan, Z.; XiaoPing, Z.; Ming, Y.; JinZhong, Y. Developing a Long Short-Term Memory (LSTM) Based Model for Predicting Water Table Depth in Agricultural Areas. *J. Hydrol. Amst.* **2018**, *561*, 918–929.
39. Zhang, J.; Zhu, Y.; Zhang, X.; Ye, M.; Yang, J. Developing a Long Short-Term Memory (LSTM) Based Model for Predicting Water Table Depth in Agricultural Areas. *J. Hydrol.* **2018**, *561*, 918–929, doi:10.1016/j.jhydrol.2018.04.065.
40. Kisi, O.; Cimen, M. A Wavelet-Support Vector Machine Conjunction Model for Monthly Streamflow Forecasting. *J. Hydrol.* **2011**, *399*, 132–140, doi:10.1016/j.jhydrol.2010.12.041.
41. Kilsdonk, R.A.H.; Bomers, A.; Wijnberg, K.M. Predicting Urban Flooding Due to Extreme Precipitation Using a Long Short-Term Memory Neural Network. *Hydrology* **2022**, *9*, 105, doi:10.3390/hydrology9060105.
42. Ahmad, M.; Al Mehedi, M.A.; Yazdan, M.M.S.; Kumar, R. Development of Machine Learning Flood Model Using Artificial Neural Network (ANN) at Var River. *Liquids* **2022**, *2*, 147–160, doi:10.3390/liquids2030010.
43. Mehedi, M.A.A.; Khosravi, M.; Yazdan, M.M.S.; Shabanian, H. Exploring Temporal Dynamics of River Discharge Using Univariate Long Short-Term Memory (LSTM) Recurrent Neural Network at East Branch of Delaware River. *Hydrology* **2022**, *9*, 202, doi:10.3390/hydrology9110202.
44. Pennypack Creek Data Collection | CUAHSI HydroShare Available online: https://www.hydroshare.org/resource/bf8e51d6a9024cc3b066fb55851d3b22/ (accessed on 1 June 2022).
45. Philadelphia Water Department Available online: https://water.phila.gov/ (accessed on 6 October 2022).
46. Mohammed, W.; Welker, A.L. Impact of Soil Compaction on Vegetated Basin Transition. In Proceedings of the Geo-Congress 2020; American Society of Civil Engineers: Minneapolis, Minnesota, February 21 2020; pp. 256–264.
47. Mohammed, W.; Welker, A.L. Hydrologic Performance of a Multicell Vegetated Basin with Different Soil and Outlet Structure Characteristics. *J. Sustain. Water Built Environ.* **2022**, *8*, 04022004, doi:10.1061/JSWBAY.0000982.
48. Sklearn.Linear_model.LinearRegression Available online: https://scikit-learn/stable/modules/generated/sklearn.linear_model.LinearRegression.html (accessed on 5 September 2022).
49. Sklearn.Linear_model.Ridge Available online: https://scikit-learn/stable/modules/generated/sklearn.linear_model.Ridge.html (accessed on 5 September 2022).
50. Sklearn.Linear_model.Lasso Available online: https://scikit-learn/stable/modules/generated/sklearn.linear_model.Lasso.html (accessed on 5 September 2022).
51. Bengio, Y.; Simard, P.; Frasconi, P. Learning Long-Term Dependencies with Gradient Descent Is Difficult. *IEEE Trans. Neural Netw.* **1994**, *5*, 157–166, doi:10.1109/72.279181.
52. Kilinc, H.C.; Haznedar, B. A Hybrid Model for Streamflow Forecasting in the Basin of Euphrates. *Water* **2022**, *14*, 80, doi:10.3390/w14010080.
53. Song, X.; Liu, Y.; Xue, L.; Wang, J.; Zhang, J.; Wang, J.; Jiang, L.; Cheng, Z. Time-Series Well Performance Prediction Based on Long Short-Term Memory (LSTM) Neural Network Model. *J. Pet. Sci. Eng.* **2020**, *186*, 106682, doi:10.1016/j.petrol.2019.106682.
54. Akatu, W.; Khosravi, M.; Mehedi, M.A.A.; Mantey, J.; Tohidi, H.; Shabanian, H. Demystifying the Relationship Between River Discharge and Suspended Sediment Using Exploratory Analysis and Deep Neural Network Algorithms. *Preprints*. **2022**, 2022110437, doi: 10.20944/preprints202211.0437.v1.
55. Zhu, X.; Khosravi, M.; Vaferi, B.; Nait Amar, M.; Ghriga, M.A.; Mohammed, A.H. Application of Machine Learning Methods for Estimating and Comparing the Sulfur Dioxide Absorption Capacity of a Variety of Deep Eutectic Solvents. *J. Clean. Prod.* **2022**, *363*, 132465, doi:10.1016/j.jclepro.2022.132465.
56. Karimi, M.; Khosravi, M.; Fathollahi, R.; Khandakar, A.; Vaferi, B. Determination of the Heat Capacity of Cellulosic Biosamples Employing Diverse Machine Learning Approaches. *Energy Sci. Eng.* **2022**, *10*, 1925–1939, doi:10.1002/ese3.1155.
57. Khosravi, M.; Tabasi, S.; Hossam Eldien, H.; Motahari, M.R.; Alizadeh, S.M. Evaluation and Prediction of the Rock Static and Dynamic Parameters. *J. Appl. Geophys.* **2022**, *199*, 104581, doi:10.1016/j.jappgeo.2022.104581.
58. Abdollahzadeh, M.; Khosravi, M.; Hajipour Khire Masjidi, B.; Samimi Behbahan, A.; Bagherzadeh, A.; Shahkar, A.; Tat Shahdost, F. Estimating the Density of Deep Eutectic Solvents Applying Supervised Machine Learning Techniques. *Sci. Rep.* **2022**, *12*, 4954, doi:10.1038/s41598-022-08842-5.