Modeling and interpreting hydrological responses of sustainable urban drainage systems with explainable machine learning methods

Yang Yang[1], Ting Fong May Chui[1]

[1]Department of Civil Engineering, The University of Hong Kong, Hong Kong SAR, China

Correspondence to: Ting Fong May Chui (maychui@hku.hk)

Abstract. Sustainable urban drainage systems (SuDS) are decentralized stormwater management practices that mimic natural drainage processes. The hydrological processes of SuDS are often modeled using process-based models. However, it can require considerable effort to set up these models. This study thus proposes a machine learning (ML) method to directly learn the statistical correlations between the hydrological responses of SuDS and the forcing variables at sub-hourly time scales from observation data. The proposed methods are applied to two SuDS catchments with different sizes, SuDS practice types, and data availabilities in the U.S. for discharge prediction. The resulting models have high prediction accuracies (NSE > 0.70). ML explanation methods are then employed to derive the basis of each ML prediction, based on which the hydrological processes being modeled are then inferred. The physical realism of the inferred hydrological processes is then compared to that would be expected based on the domain-specific knowledge of the system being modeled. The inferred processes of some models, however, are found to be physically implausible. For instance, negative contributions of rainfalls to runoffs have been identified in some models. This study further empirically shows that an ML model's ability to provide accurate predictions can be uncorrelated with its ability to offer plausible explanations to the physical processes being modeled. Finally, this study provides a high-level overview of the practices of inferring physical processes from the ML modeling results and shows both conceptually and empirically that large uncertainty exists in every step of the inference processes. In summary, this study shows that ML methods are a useful tool for predicting the hydrological responses of SuDS catchments and the hydrological processes inferred from modeling results should be interpreted cautiously due to the existence of large uncertainty in the inference processes.

1 Introduction

Sustainable urban drainage systems (SuDS), also known as low impact development practices, green infrastructure, and sponge city, are decentralized stormwater management practices that aim to promote onsite infiltration, storage, evapotranspiration, and stormwater reuse (Fletcher et al., 2015; Jones and Macdonald, 2007). SuDS can effectively improve stormwater runoff quality, reduce runoff volume, and restore natural hydrological regimes (Selbig et al., 2019; Trinh and Chui, 2013; Zhou, 2014). Commonly used SuDS include bioretention cells, green roofs, porous pavement, and rain barrels (Gimenez-Maranges et al., 2020; Charlesworth, 2010).

A number of numerical modeling methods have been adopted or developed to predict the hydrological performance of SuDS and understand the involved hydrological processes (Liu et al., 2014; Elliott and Trowsdale, 2007). The simplest methods are perhaps those developed based on empirical equations for assessing the drainage impact of different land-use types. For instance, the rational method and SCS runoff curve number method are modified and used in Montalto et al. (2007) and Damodaram et al. (2010) to study the effectiveness of SuDS at catchment scales. Empirical equation-based methods can be useful in preliminary designs to rapidly estimate some key performance metrics of SuDS. However, these methods may poorly reflect detailed SuDS design variations (Fassman-Beck et al., 2016).

Process-based models are another approach to modeling SuDS, in which physically-based or empirical equations are used to characterize the involved hydrological processes. SuDS are typically represented in process-based models as hydrological functional units, whose properties are defined using a set of parameters. Commonly used models, including SWMM and MUSIC, are reviewed in Eckart et al. (2017) and Elliott and Trowsdale (2007).

The application of process-based models, however, faces several challenges. First, it may require considerable effort to set up a process-based model for SuDS, as not all the required parameters are measurable or can be measured at a reasonable cost. For example, in SWMM, the initial soil moisture deficit parameter of SuDS is often determined through calibration (Rosa et al., 2015). Second, some complex hydro-environmental processes of SuDS and surrounding environments are difficult to model using existing models. For instance, SWMM does not account for macropore flow in the SuDS soil layer (Niazi et al., 2017), and models that assess the performance of SuDS in shallow groundwater environments (Zhang and Chui, 2019) and cold climates (Johannessen et al., 2017) are limited. Third, the assumptions used in process-based models may be invalid in some cases due to unknown issues related to construction, maintenance, or physical property changes during a SuDS' service life (Yong et al., 2013).

It may be useful to model directly the statistical correlations between the random variables that describe the states of SuDS catchments. The resulting statistical models may be adopted for solving various prediction tasks and used as references to assess the prediction accuracy of process-based models. These models may be derived using machine learning (ML) methods, which aim to learn the statistical correlations between random variables from observation data (Solomatine and Ostfeld, 2008). Terms that are closely related to ML include data-driven modeling, predictive modeling, and statistical learning.

ML methods have been widely used in various fields in hydrology (Maier and Dandy, 2000). However, they have only been used in a few SuDS-related studies. For instance, linear regression methods were used in Eric et al. (2015), Hopkins et al. (2020), and Khan et al. (2013) to predict the hydrological effectiveness of SuDS, such as runoff volume reduction, based on factors such as inflow volume, antecedent soil moisture content, and SuDS implementation levels. Li et al. (2019) used neural networks models to predict the peak flow and runoff volume of runoff events of a SuDS site based on rainfall event characteristics. The studies mentioned above focused on predicting the long-term or rainfall event-level hydrological performance of SuDS. However, there is currently insufficient literature on the application of ML methods to model the temporal evolution of the hydrological responses of SuDS at regular time steps, e.g., daily, hourly, or sub-hourly. Yang and Chui (2019) showed that ML methods, such as deep learning methods and random forest methods, are useful for predicting

the runoff response of SuDS at sub-hourly time scales, provided that the model's input variables are appropriate. However, a method to derive these input variables was not described in their study.

The lack of popularity of ML methods in SuDS-related studies may be explained by several factors. First, ML methods are not applicable when observation data of the variables of interest are unavailable. Second, modeling the hydrological responses of SuDS at fine temporal scales requires a high-dimensional hydrometeorological time series to be used as input, which can be challenging for ML methods that are not specifically designed for modeling sequence data (Nielsen, 2019). Additionally, ML methods may also not offer clear advantages over equation-based methods when applied to study the performance of SuDS at the rainfall event level. Third, ML models are usually trained to capture the statistical correlations between random variables without or with little consideration of the involved physical processes, thus they may be considered less useful for understanding the physical processes compared to process-based models.

Therefore, to promote the application of ML methods in SuDS related studies, one must show that ML methods can provide accurate predictions for various tasks and that the involved hydrological processes can be interpreted. While it is straightforward to apply specific ML methods to solve prediction tasks, there is currently insufficient research into interpreting the hydrological processes learned by ML models.

Several studies in hydrology explained ML models using methods adopted from explainable artificial intelligence (XAI), which is an emerging field of ML that aims to make ML modeling results more understandable to humans (Bojanowski et al., 2018). Commonly used XAI methods for understanding the functioning of ML models include transparent ML models and post-hoc explainability techniques (Barredo Arrieta et al., 2020). Transparent ML models refer to those with structures that are directly understandable to humans, which include linear regression models, decision trees, and K-nearest neighbors. These models have been frequently adopted in hydrology for understanding the correlations between random variables or the basis of specific predictions (Solomatine and Dulal, 2003; Wani et al., 2017). Post-hoc explainability techniques aim to explain ML models that are not transparent. For instance, in hydrology, the integrated gradients method (Sundararajan et al., 2017) has been used in Kratzert et al. (2019) to understand the contribution of meteorological input at different time steps to streamflow discharge prediction in neural networks, the permutation feature importance method has been used in Schmidt et al. (2020) to assess the importance of the predictors for flood magnitude prediction in various ML models, and the SHAP method (Lundberg and Lee, 2017) has been used in Starn et al. (2021) to identify the factors affecting groundwater residence time distribution predictions in XGBoost models (Chen and Guestrin, 2016).

Most of the current hydrology literature uses post-hoc explainability techniques to test whether an ML model makes right predictions for the right reasons, where a model is generally considered more trustworthy if it can generate predictions in a way that is consistent with our knowledge of the system being modeled. Here, the term trustworthy is defined broadly as the quality of a model to provide predictions that can be trusted (Morton, 1993). The current applications essentially test the patterns learned by ML models against that would be expected from the domain-specific knowledge of the system being modeled (Yang and Chui, 2021), and the test results are then used as an indicator of a model's trustworthiness. This approach, however, may be challenged by the fact that ML models can uncover hidden patterns in data that make no intuitive sense to

humans (Ilyas et al., 2019). Thus, the quality of a model to provide accurate predictions and plausible explanations to the physical processes may be uncorrelated. Rudin (2019) further suggests that the post-hoc explainability techniques themselves are uncertain and approximated because inaccurate representations of the original model may be adopted in deriving the explanations, and similar views on the uncertainties in explanation are also reported in Chen et al. (2020) and Sundararajan and Najmi (2020).

Therefore, in hydrological studies, it is meaningful to ask whether post-hoc explainability techniques and ML models can provide physically plausible explanations to the processes of the system being modeled and whether a model's abilities to provide accurate predictions and plausible explanations are correlated. This study aims to investigate these questions by examining the ML models that are trained to predict hydrological responses of SuDS catchments at sub-hourly time scales, and through which the applicability of ML methods to modeling SuDS catchments is also assessed.

2 Methods and materials

2.1 Training and testing machine learning models

2.1.1 Modeling hydrological responses of SuDS using machine learning methods

Let random variable Y_t denote the hydrological response of a SuDS catchment at time step t and random vector $\mathbf{X_t}$ denote the time series of the hydrometeorological conditions and other factors measured on and before time step t.

$$\mathbf{X_t} \coloneqq [P_t, P_{t-1}, P_{t-2}, \dots, E_1, E_2, \dots, E_k], \tag{1}$$

where P_{t-i} is the rainfall depth recorded at time step $t-i$, and E_1 through E_k represent k measurements of the other variables. P_{t-0} is written as P_t for convenience.

It is assumed that Y_t can be written as an unknown function of $\mathbf{X_t}$, which can be approximated by functions learned by ML algorithms from observation data of $\mathbf{X_t}$ and Y_t. A feature engineering process is commonly involved in the learning process, in which $\mathbf{X_t}$ is converted to lower-dimensional representations using a function g such that the mapping between $g(\mathbf{X_t})$ and Y_t can be learned more easily by ML algorithms (Kuhn and Johnson, 2019). Y_t can then be estimated using $\widehat{Y_t}$, which is computed following

$$\widehat{Y_t} = f_\theta\left(g_\varphi(\mathbf{X_t})\right), \tag{2}$$

where f is a function learned by an ML algorithm, and φ and θ are parameters of g and f. Figure 1 illustrates the processes for deriving the prediction for an input sample $\mathbf{x_t}$.

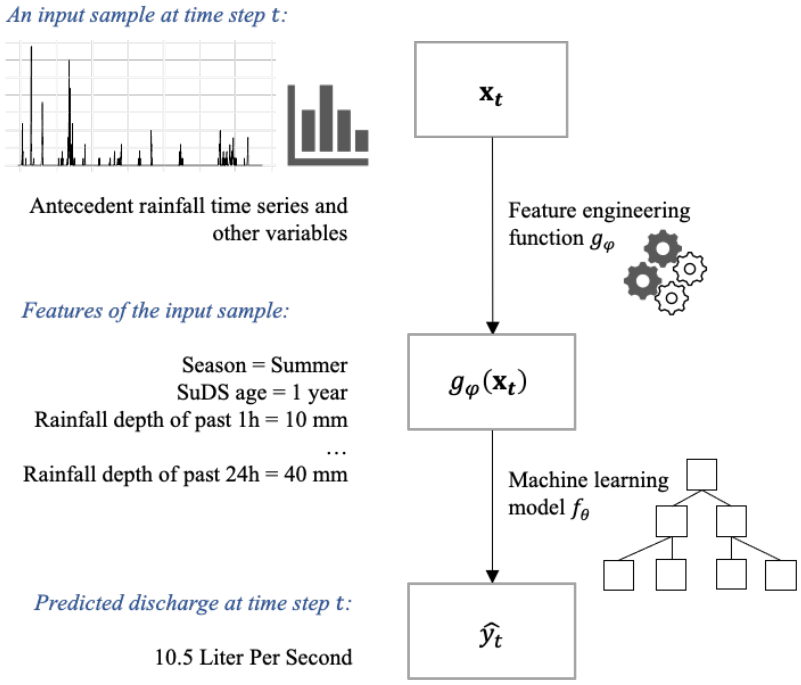

Figure 1 Illustration of the prediction generation process for an input sample $\mathbf{x}_t$.

The goal of ML is then to identify the optimal parameter values θ^* and φ^* that minimize the expected loss ℓ over the data distribution $p_d(\mathbf{X_t}, Y_t)$, as shown in

$$(\theta^*, \varphi^*) = \underset{\theta,\varphi}{\operatorname{argmin}} E_{(\mathbf{X_t},Y_t)\sim p_d(\mathbf{X_t},Y_t)} \ell\left(f_\theta\left(g_\varphi(\mathbf{X_t})\right), Y_t\right) \quad (3)$$

As $p_d(\mathbf{X_t}, Y_t)$ are unknown, the expectation is often approximated by averaging the losses computed for a set of observed samples $(\mathbf{x_t}, y_t)$.

2.1.2 Feature engineering methods

Gauch et al. (2021) showed that the hydrometeorological time series recorded in the long-term past can be represented using a coarser temporal resolution in ML models built for rainfall-runoff modeling without deteriorating their prediction accuracy. This study adopts a similar approach to represent a rainfall time series by aggregated rainfall depths recorded during different intervals, in which rainfall time series recorded between time steps $t-a$ and $t-b$ is represented by a rainfall depth feature $D_{t-a,t-b}$,

$$D_{t-a,t-b} = \sum_{i=a}^{b} P_{t-i}, \quad (4)$$

However, an approach to optimally define the set of (a, b) pairs to create rainfall depth features is not known *a priori*.

This study proposes a simple method to systematically select cut points along the time axis, which form a series of intervals for defining (a, b) pairs. As shown in Figure 2, the selection of cut points is controlled by three hyperparameters, m, l, and n. (1) m: a cut point is placed between time steps $t - m$ and $t - m - 1$ such that the rainfall data recorded prior to time step $t - m$ are considered irrelevant for predicting Y_t. (2) l: the rainfall data recorded between time steps $t - l$ and $t - 0$ are considered to be most relevant for predicting Y_t, so that cut points are placed around each time step within this interval. (3) n: $n - 1$ cut points are placed between time steps $t - l - 1$ and $t - m$, such that the neighboring cut points correspond to n intervals whose lengths roughly form an arithmetic sequence. After the (a, b) pairs having been defined, a rainfall depth feature $D_{t-a,t-b}$ is then created for every interval formed by two neighboring cut points.

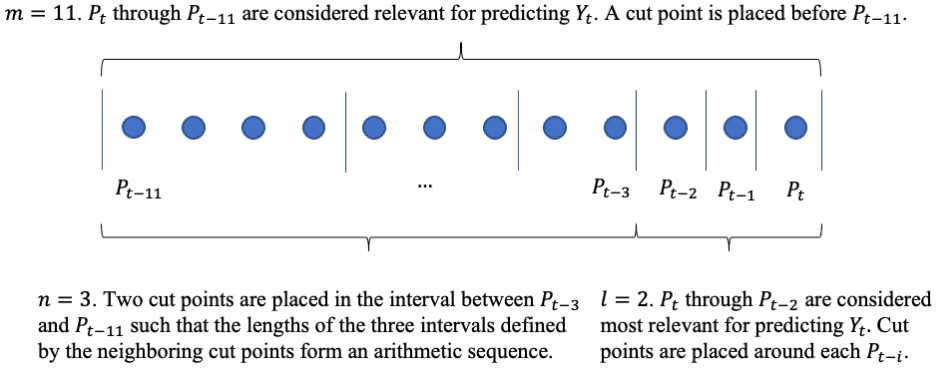

Figure 2 Illustration of the method to place cut points along the time axis.

Representing a rainfall time series using a set of $D_{t-a,t-b}$ can reduce the dimensionality of data at the cost of losing information regarding the temporal distribution of rainfall. In this method, fewer cut points are selected for rainfalls in the long-term past (e.g., a few days ago), which is based on the assumption that they are less important for predicting Y_t. This is reasonable considering the relatively fast response time of SuDS (DeBusk et al., 2011). Similarly, some of the environmental variables $[E_1, E_2, \ldots, E_k]$ may be less important for predicting Y_t, which can be filtered out during the feature engineering process. In this study, whether or not to include E_i is controlled by a Boolean variable, and k such variables are used.

In this study, the optimal values of the feature engineering hyperparameters, m, l, n, and the k Boolean variables, are determined using resampling and Bayesian optimization methods as described below.

2.1.3 XGBoost algorithm

This study adopts the gradient-boosted trees algorithm (Friedman, 2001) to train ML models. In particular, the XGBoost (Chen and He, 2020) software library is used. XGBoost is selected for its improved regularization methods, high computational efficiency, and ability to achieve state-of-the-art results on various ML tasks (Nielsen, 2016; Chen and He, 2020; Chen and Guestrin, 2016). A detailed introduction to XGBoost can be found in Chen and Guestrin (2016) and Mitchell and Frank (2017).

A gradient boosted trees model G is an ensemble of decision trees, in which $\hat{y}_i$, the prediction for an input sample $\mathbf{x}_i$, is the sum of the predictions of individual trees (Chen and Guestrin, 2016), given as

$$\hat{y}_i = G(\mathbf{x}_i) = \sum_{k=1}^{K} f_k(\mathbf{x}_i), f_k \in \mathcal{F}, \quad (5)$$

where f_k is a decision tree that maps input samples to the values stored at the tree leaves, K is the number of decision trees (also known as the number of boosting iterations), and $\mathcal{F}$ is the functional space of all possible regression trees. For a given dataset, the structure of the trees, including the splitting criteria and the values stored at the leaves, is learned automatically using XGBoost.

There are a number of hyperparameters used in XGBoost for controlling model structure and the learning behaviors during training, e.g., the number of boosting iterations and maximum tree depth. A complete list of the XGBoost hyperparameters can be found in the software documentation (Chen and He, 2020). In this study, the XGBoost hyperparameters are optimized together with feature engineering hyperparameters using resampling and Bayesian optimization methods as described below.

2.1.4 Resampling methods and Bayesian optimization for training and testing machine learning models

In this study, the effectiveness of the feature engineering and XGBoost algorithm are evaluated on different datasets of observed $(\mathbf{x}_t, y_t)$ samples collected at different SuDS sites. For each dataset, the evaluation is performed by randomly splitting the dataset into a series of training and test subsets. For each such split, during the hyperparameter optimization phase, the training set is further split into a series of smaller training and validation datasets. Then, multiple models with different feature engineering and XGBoost hyperparameters are trained on the smaller training datasets, and the quality of a set of hyperparameters is measured by the prediction accuracy of the resulting model on the validation datasets, and the optimal hyperparameters are then identified. During the model evaluation phase, the optimal hyperparameters are then used to fit a model on the training dataset (which includes both the smaller training and validation datasets), and the resulting model is evaluated using the test dataset. Apparently, the assessment results are affected by how the dataset is split, thus the hyperparameter optimization and model evaluation processes are repeated for various splits in this study for some numerical experiments. More information on resampling methods can be found in Kuhn and Johnson (2013) and Hastie et al. (2009).

During the hyperparameter optimization phase, the candidate hyperparameters to be assessed are proposed by Bayesian optimization methods, which are sample-efficient algorithms for solving black-box optimization problems (Shahriari et al., 2016). Bayesian optimization methods are commonly used in ML for hyperparameter optimization (Snoek et al., 2012). The decision variables in the optimization problems are the hyperparameters, and the quantity to be optimized is the prediction accuracy on the validation datasets. An introduction to Bayesian optimization can be found in Frazier (2018).

2.2 Interpreting model structures and inferring hydrological processes learned by machine learning models

2.2.1 Interpreting the basis of each prediction

Understanding why a specific prediction is made by an ML model can be useful for understanding the relationships between various variables captured by the model. In this study, XGBoost uses decision trees as its base learner. Although each decision tree can be considered as a transparent ML model (as the rules used for making predictions can be understood easily by humans), it can be challenging to directly interpret the prediction generation process of an XGBoost model as many trees can be used in it. Therefore, post-hoc explainability techniques, such as the *gain, cover,* and *frequency* metrics, are commonly used for understanding the structure of XGBoost models.

The gain of a feature is its relative contribution to the model as measured by the total gain of the feature's splits. It can be roughly regarded as a feature's contribution to prediction accuracy improvement (Chen and He, 2020). The cover of a feature is the relative number of training samples related to the feature's splits (Chen and He, 2020). The frequency of a feature is the relative number of times that this feature has been used in tree splits (Chen and He, 2020). The three metrics are global feature importance measures, as they reflect the overall contribution of a feature to an XGBoost model for making various predictions (Guidotti et al., 2019; Ahmad et al., 2018). However, these metrics can be irrelevant for understanding the basis of a specific prediction, which is a task that requires local explanation methods.

This study adopts a local feature attribution method to quantify the contribution of each feature to the prediction made for a specific input sample (Janzing et al., 2019). The SHAP (SHapley Additive exPlanations) method proposed by Lundberg and Lee (2017) is used in this study. The SHAP value of a feature for a specific input sample can be considered as the marginal contribution of this feature to the predicted value compared to the mean predictions for all samples. SHAP values satisfy a series of desired properties. For instance, the sum of the SHAP value assigned to each feature equals the difference between the predicted value and the mean prediction for all samples, and the features that do not change the expected prediction are assigned with a SHAP value of 0 (Lundberg et al., 2020). The SHAP values can be computed using the following steps.

Let the real-valued function f of N-dimensional random variable $\boldsymbol{X}$ be the ML model to be explained and $\mathbf{x} \coloneqq (x_1, x_2, \dots, x_N)$ be an observed sample of $\boldsymbol{X}$. $\emptyset_i$, the SHAP value of x_i, is computed as

$$\emptyset_i = \frac{1}{N!}\sum_{R\in\mathcal{R}}[v(S^R \cup i) - v(S^R)], \tag{6}$$

where, R is a random permutation of the N features, $\mathcal{R}$ is the space of all feature permutations, S^R is the set of features that are located before feature i in permutation R, and $v: S \in \mathcal{P}(N) \rightarrow \mathbb{R}$ is a set function that maps every subset of the N features (i.e., each member of the power set $\mathcal{P}$ of all N features) to a real number, and v is known as the value function.

Therefore, $\emptyset_i$ can be interpreted as the expected marginal contribution to v of feature i (i.e., $v(S^R \cup i) - v(S^R)$) in a random permutation of the N features. Equation 6 is developed based on the Shapley value used in game theory, more information on which can be found in Shapley (1953) and Osborne and Rubinstein (1994).

In the SHAP method, v is the expected prediction of f when some features are "missing". v may be defined in various ways. Lundberg and Lee (2017) define v using the observational conditional expectation, which is the expected value of $f(\boldsymbol{X})$ when the feature values of $\boldsymbol{X}$ for a set of features S are known, as in

$$v(S) = E[f(\boldsymbol{X})|\boldsymbol{X}_S = \mathbf{x}_S] , \quad (7)$$

Janzing et al. (2019) and Lundberg et al. (2020) defined v using the interventional conditional expectation, as in

$$v(S) = E[f(\boldsymbol{X})|do(S)] , \quad (8)$$

where $do(S)$ represents an intervention that sets the feature values of $\boldsymbol{X}$ in S to $\mathbf{x}_S$. The SHAP values derived using Equations 7 and 8 are respectively termed *observational SHAP values* and *interventional SHAP values*. The observational SHAP value of x_i generally measures the value of knowing x_i to predict the outcome, and the interventional SHAP value of x_i corresponds to the expected changes in the model prediction when the feature X_i is set to the x_i.

Chen et al. (2020) suggested that both observational and interventional SHAP values are useful. They claimed that the observational SHAP values are "true to the data" because they are effective in identifying the true correlations between the features and the outcome of interest, whereas the interventional SHAP values are "true to the model" because they do not credit the features that are unused by the model. The observational SHAP values are used in most places of this paper as they are less computationally expensive than the interventional SHAP values. The TreeSHAP methods proposed in Lundberg et al. (2020) are used in this study to compute both SHAP values.

For a given input sample $\mathbf{x_t}$, the SHAP value assigned to a rainfall depth feature $d_{t-a,t-b}$ can be further distributed among the rainfall recorded at each time step between time steps $t-a$ and $t-b$. The SHAP value $\emptyset_{D_{t-a,t-b}}(\mathbf{x_t})$ can be assigned to the rainfall recorded at time step $t-k$ proportional to its depth p_{t-k}, where $a \le k \le b$. Thus, $\tau_k(\mathbf{x_t})$, the SHAP value assigned to p_{t-k} if the rainfall depth recorded between time steps $t-a$ and $t-b$ (denoted by $d_{t-a,t-b}$) is not 0 can be computed using

$$\tau_k(\mathbf{x_t}) = \frac{p_{t-k}}{d_{t-a,t-b}} \emptyset_{D_{t-a,t-b}}(\mathbf{x_t}) , \quad (9)$$

The processes for quantifying the contribution of each feature of an input sample $\mathbf{x_t}$ to model prediction and distributing the contributions to the rainfall of each time step are illustrated in Figure 3a.

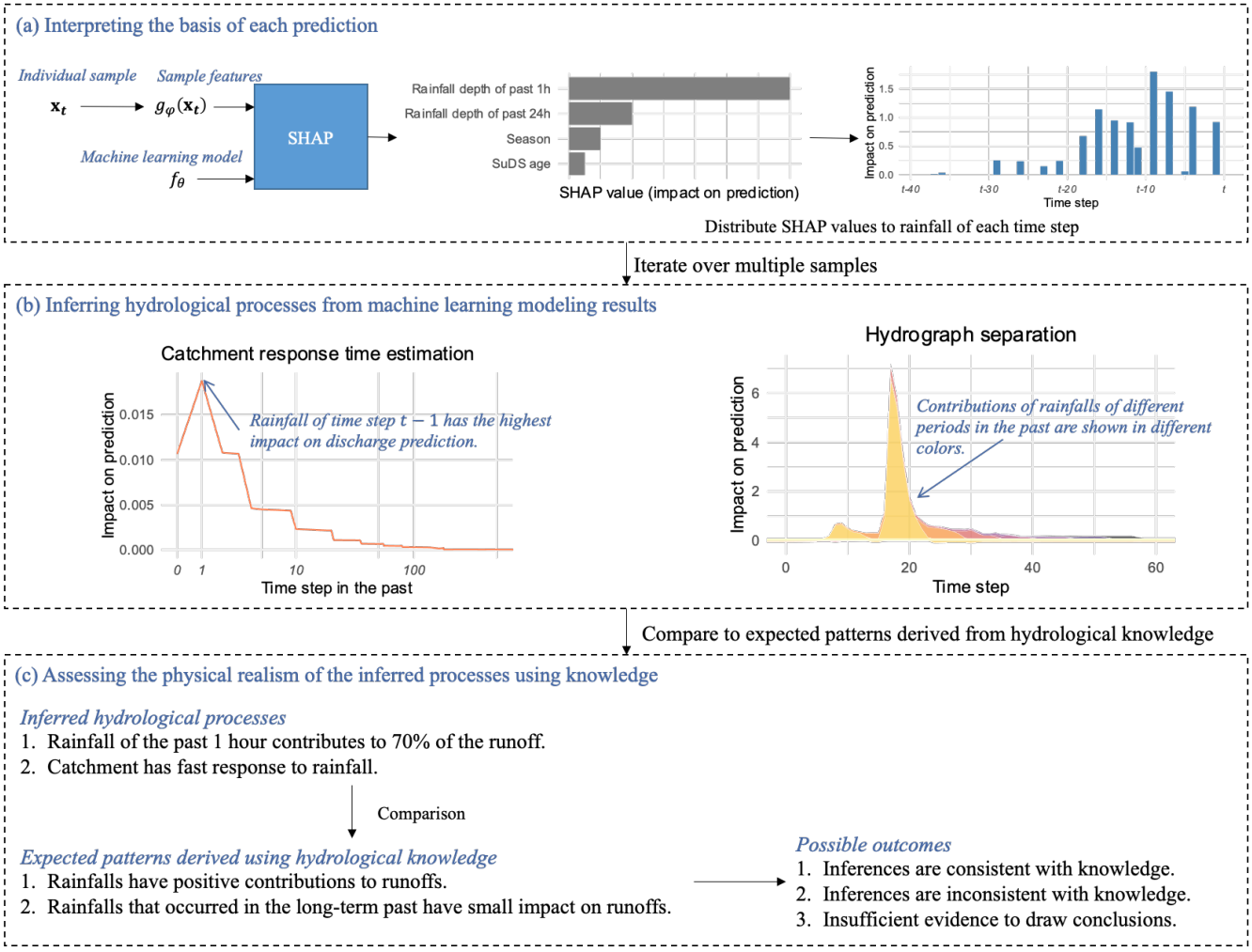

Figure 3 (a) Illustration of the processes for quantifying the contribution of each feature of an input sample $\mathbf{x}_t$ and distributing the contribution to the rainfall of each time step. (b) Examples of inferring the hydrological processes being modeled based on the basis of each prediction. (c) Examples of comparing the inferred hydrological processes to the expected patterns of the processes derived from domain-specific knowledge of the system being modeled.

2.2.2 Inferring hydrological processes from machine learning modeling results

The process of inferring the hydrological processes being modeled involves mapping from the explanations on the model structure to some imaginary catchments that are likely to possess the characteristics that are consistent with the explanations. For instance, if the explanations indicate that the discharge predictions are strongly controlled by the rainfalls in the long-term past, then the processes being modeled are likely to correspond to the processes of catchments with long-term memory effects. However, the mapping processes are inherently subjective and incomplete, discussions of these characteristics are presented in Section 3.6. This section introduces the methods to map the explanations to imaginary catchments, and more discussions on the limitations are given when the case study results are presented.

In this study, for each $\mathbf{x_t}$, $\tau_k(\mathbf{x_t})$ is computed for p_{t-k} of each time step (Equation 9). These $\tau_k(\mathbf{x_t})$ values quantitively describe the associations between rainfall and the hydrological response across various time steps, which is useful for inferring the catchment's hydrological processes. For instance, if the predictions are found to be mostly controlled by recent rainfalls, then the processes being modeled can correspond to that from a small catchment with a fast response to rainfalls. The $\tau_k(\mathbf{x_t})$ values can be useful for hydrograph separation. That is, a predicted hydrograph can be decomposed to sub-hydrographs associated with rainfalls that occurred in different periods based on their contribution to the predicted discharge, which is different from the current practices that mostly decompose a hydrograph based on the origin of the runoffs, such as baseflow and overland flow (Pelletier and Andréassian, 2020). The implications of using this method are discussed in Section 3.2.

The SHAP values of multiple samples can be analyzed collectively to obtain a global understanding of the model structure and the system being modeled (Lundberg et al., 2020). This study thus computes the expected $\tau_k(\mathbf{x_t})$ values for multiple $\mathbf{x_t}$ in a set S when k is fixed to understand the average association between p_{t-k} and $f(\mathbf{x_t})$ using

$$E_{\mathbf{x_t}\in S}\big(\tau_k(\mathbf{x_t})\big) = \frac{\sum_{\mathbf{x_t}\in S}\tau_k(\mathbf{x_t})}{|S|}, \quad (10)$$

where $|S|$ is the number of elements of S. When S contains all the $\mathbf{x_t}$ samples, the expectation $E_{\mathbf{x_t}\in S}\big(\tau_k(\mathbf{x_t})\big)$ then approximately describes the overall association between p_{t-k} and $f(\mathbf{x_t})$ learned by the model.

SHAP values can be negative, which will result in negative $\tau_k(\mathbf{x_t})$ values. To avoid canceling out positive and negative $\tau_k(\mathbf{x_t})$ values when computing the expectations, the absolute value of $\tau_k(\mathbf{x_t})$ may be used. The quantity of $|\tau_k(\mathbf{x_t})|$ can be interpreted as the importance of p_{t-k} for computing $f(\mathbf{x_t})$. The expected value of $|\tau_k(\mathbf{x_t})|$ for the $\mathbf{x_t}$ samples in a set S be computed using

$$E_{\mathbf{x_t}\in S}(|\tau_k(\mathbf{x_t})|) = \frac{\sum_{\mathbf{x_t}\in S}|\tau_k(\mathbf{x_t})|}{|S|}, \quad (11)$$

Similarly, for a given $\mathbf{x_t}$, the contribution of rainfalls recorded between time steps $t-a$ and $t-b$, $T_{a,b}(\mathbf{x_t})$, can be simply computed as

$$T_{a,b}(\mathbf{x_t}) = \sum_{i=a}^{b}\tau_i(\mathbf{x_t}) \quad (12)$$

Figure 3b gives two examples of inferring hydrological processes from ML modeling results.

2.2.3 Assessing the physical realism of the inferred processes using knowledge

The inferred hydrological processes may be tested in terms of their physical realism. The premise of this test is that if an ML model can provide physically plausible explanations to the processes it models, then its predictions can generally be considered more trustworthy. That is, this test concerns whether the right predictions are made for the right reasons (Kirchner, 2006). The justification of this method is examined in Section 3.6.

In the proposed assessment method, the inferred hydrological processes are compared to the hydrological processes that would be expected based on the domain-specific knowledge of the system being modeled. In another word, whether the inferred hydrological processes are physically plausible are evaluated. In an assessment, the qualitative or quantitative

descriptions of the inferred processes and that derived from domain-specific knowledge are used in the comparison. For instance, a small urban catchment is modeled, and it is expected to have a fast response to rainfalls. If the inferred hydrological processes correspond to that from a catchment with a long-term memory effect, then the ML model can be considered unreliable. Generally, three possible outcomes are expected in such an assessment.

1. *Consistent*. These are cases when the inferred hydrological processes are physically plausible according to the domain-specific knowledge of the system being modeled. The term "consistent" is used, rather than "correct" or "valid", is to reflect that the assessment process involves a comparison to some basis derived from our knowledge of the system being modeled, which might be subjective and incomplete. The term is used following Yang and Chui (2021).

2. *Inconsistent*. These are cases when the inferred hydrological processes are physically implausible according to the domain-specific knowledge of the system being modeled.

3. *Insufficient evidence to draw conclusions*. These are cases when definitive conclusions cannot be drawn, which may be caused by that the requirements for the inferred processes to be considered consistent are too specific or too general. For example, assume that the inferred hydrological processes indicate that the catchment has a small surface area (i.e., a qualitative description), and the time of concentration of the catchment being modeled is known from previous studies and is used as the assessment criterion (i.e., a quantitative description). In this context, it is impossible to determine the consistency between the two descriptions unless more evidence regarding the conversion between catchment scale and time of concentration is collected. The inability to draw a definitive conclusion can also be caused by the lack of knowledge of the processes being modeled. For instance, it is difficult to identify the expected hydrological behaviors for ungauged natural catchments.

Figure 3c illustrates the processes for assessing the physical realism of the inferred hydrological processes.

2.3 Case studies

2.3.1 Study sites

Two SuDS sites with different drainage areas, SuDS practice types, and data availabilities are examined in this study. Study site 1 is located on Washington Street, Geauga County, Ohio, U.S. (hereinafter referred to as "WS"). Multiple types of SuDS were built in WS to treat stormwater runoffs generated by a nearby commercial building and parking lot, as shown in Figure 4a (Darner et al., 2015). Runoffs from approximately half of the commercial building roof (i.e., an impervious area of 316 m^2) drain into a rain garden with a surface area of 37 m^2. The 762 m^2 parking lot was constructed using porous pavements to allow infiltration.

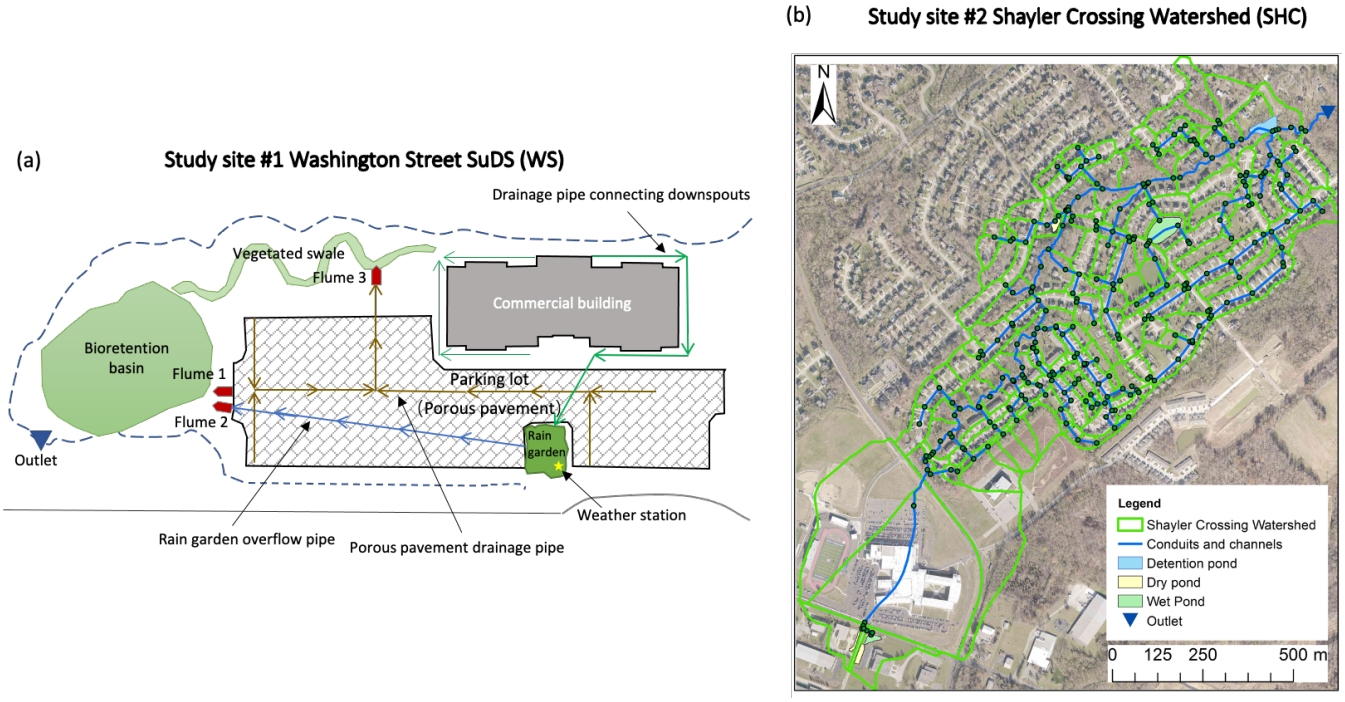

Figure 4 (a) Layout of the SuDS and monitoring network on the Washington Street site (WS), Geauga County, Ohio, U.S. This figure is adapted from Darner et al. (2015). (b) Map of the Shayler Crossing Watershed (SHC). The subcatchment boundaries and drainage system shown on the map are defined by Lee et al. (2018a).

Study site 2 is the Shayler Crossing Watershed (SHC) in Clermont County, Ohio, U.S., as shown in Figure 4b. SHC is a sub-watershed of the East Fork Little Miami River Watershed. The drainage area of SHC is approximately 0.92 km^2 (Hoghooghi et al., 2018) and the land use type is primarily residential. The drainage system of SHC consists of conduits, channels, detention ponds, dry ponds, and wet ponds (Lee et al., 2018a). In SHC, stormwater runoff generated by indirectly connected impervious areas (e.g., sidewalks) is treated by the nearby pervious areas, which are termed buffering pervious areas and have similar functions to grass filter strips (Lee et al., 2018b). SHC represents a typical residential area in the U.S. and is thus selected to test the applicability of the proposed ML methods in modeling small urban catchments.

In WS, a 10-min-resolution rainfall-discharge time series is available from on-site monitoring between 2009 and 2013. The outflow from WS was collected and measured by three flumes. Flumes 1, 2, and 3 respectively collect the surface runoffs from the parking lot, overflow from the surface layer of the rain garden, and underdrain flows from the parking lot. The onsite monitoring was conducted by the United States Geological Survey (USGS), and more details of the monitoring work can be found in Darner and Dumouchelle (2011) and Darner et al. (2015). In SHC, 10-min-resolution rainfall time series from 2009–2010 is available, in addition to a 10-min-resolution discharge time series measured at the outlet between July and August 2009 by the U.S. Environmental Protection Agency. The dataset used in this study is the same as in Lee et al. (2018a) and Lee et al. (2018b), in which more details on the dataset can be found.

It can be challenging to set up process-based models for both sites. In WS, the physical properties and exact design of the different drainage system elements are not precisely known (Darner et al., 2015). For instance, the rain garden is not isolated from the gravel storage layer of the porous pavements, however, the exact flow conditions in the storage layer are unknown. In SHC, the main challenge lies in the heavy workload and uncertainties in estimating the model parameters that characterize the complex drainage system. For example, to accurately represent the drainage processes, SHC should be divided into multiple subcatchments connected by a drainage network, and each subcatchment should be further subdivided into multiple subareas, such as directly and indirectly connected impervious subareas (Lee et al., 2018a). The task of the sub-area division, however, requires substantial effort, considering the relatively large number of subcatchments that are involved.

2.3.2 Numerical experiments

Rainfall-runoff models are built for both SHC and WS using ML methods. In WS, the output variable is the flow rate of the total runoff collected by the three flumes recorded at regular 10-min intervals during the warm season (i.e., April to October) (Darner et al., 2015). The input variables include the 10-min-resolution rainfall time series recorded prior to runoff, the month in which the runoff occurs (optional), and the accumulative rainfall depth recorded since the beginning of monitoring (optional). The optional features are considered to account for the possible time-varying performance of the SuDS during its service life (Yong et al., 2013) and the potential seasonality of the SuDS hydrological properties (Muthanna et al., 2008). Whether the two sets of optional features should be included is controlled by two binary feature engineering hyperparameters, and the other three feature engineering hyperparameters are m, l, and n, as described in Section 2.1.2. The ranges of the hyperparameter values are listed in Table 1, and their optimal values are determined using Bayesian optimization methods. The rainfall-discharge data collected between 2010 and 2013 by USGS are used in this study. A total of 142 independent rainfall events are identified using a 24-h dry spell threshold (Guo and Senior, 2006). A nested cross-validation (CV) resampling procedure is implemented, in which 5-fold CV is used for both the inner and outer CV iterations. The folds are created using a rainfall event-grouped stratified sampling method (Zeng and Martinez, 2000), i.e., data associated with the same rainfall event are grouped into the same fold, and the peak discharge of the rainfall events in each fold roughly follows the same distribution. This is to prevent data leakage and ensure that the data in each fold are representative (Kuhn and Johnson, 2013). In general, each outer CV iteration can be considered as an experiment to assess the effectiveness of an ML method using a specific split of the dataset, and its associated inner CV iterations are considered as procedures to derive the model to be evaluated on the test dataset.

Table 1 Hyperparameter values considered for the two study sites.

	Study site #1 WS	Study site #2 SHC
Candidate feature engineering	m is a random integer between 144 and 1440, l is a random integer between 1 and 36, and n is a random integer between 2 and 36. The first term of the arithmetic sequence of the interval lengths between time steps	m, l, and n are integers and their ranges are the same as that for study site #1.

hyperparameter values	$t-l-1$ and $t-m$ is 2. The inclusion or exclusion of the accumulative rainfall depth and the occurring month of the runoff event is controlled by two binary variables.	
Candidate XGBoost hyperparameter values	eta is a real number between 0.005 and 0.1, max_depth is an integer between 2 and 10, min_child_weight is an integer between 1 and 10, $subsample$ is a real number between 0.20 and 1, $colsample\_bytree$ is a real number between 0.2 and 1, and $gamma$ is a real number between 0 and 10. The maximum value of $nrounds$ is 5000, and its optimal value is determined using an early stopping criterion that the training stops if there is no improvement in validation accuracy for 20 consecutive rounds.	Same as study site #1.

In SHC, the output variable is the watershed outlet discharge measured at 10-min intervals, and the input variable is the rainfall time series recorded before the discharge measurement. Only two months of runoff data are available in this study. The nested CV procedure is not used due to the small dataset size; instead, the dataset is split into training, validation, and test datasets that each contains at least one large runoff event. The Bayesian optimization methods are then used to identify the optimal hyperparameters (as shown in Table 1) that minimize the prediction error on the validation dataset when the model is trained on the training dataset. The training and validation datasets are then combined, and the ML methods with the optimal hyperparameters are applied. The resulting models are then tested on the test dataset.

For each site, the resampling and hyperparameter optimization methods are also applied to train linear regression models, which are used as a baseline for evaluating XGBoost models. The only difference in the training processes between the two model types is that only the feature engineering hyperparameters are used when fitting linear regression models to the data. For SHC, the process-based model developed by Lee et al. (2018a) is also compared with the ML models built in this study. Their model is built using SWMM software, in which SHC is divided into 191 subcatchments and the drainage processes in each subcatchment are characterized using various parameters. The prediction accuracies of different types of models are then compared for each site.

The proposed method is then applied to explain the basis of each prediction for the two sites, i.e., for each discharge prediction, the contribution of rainfall of each time step (i.e., $\tau_k(\mathbf{x_t})$) is computed. Both the observational and interventional SHAP values are used in the derivation, which results in two versions of the $\tau_k(\mathbf{x_t})$ values. The following experiments on inferring hydrological processes from ML modeling results are conducted based on the $\tau_k(\mathbf{x_t})$ values.

1. The predicted hydrographs are decomposed into multiple hydrographs associated with the rainfalls recorded between the past 0–1 h, 1–2 h, and so on using Equation 12. Whether the hydrograph separation method can generate physically plausible results is examined using a few simple hydrological principles, which include (a) rainfalls have positive

contributions to runoffs and (b) runoffs in small urban catchments are mostly contributed by rainfalls that occurred in the recent past.

2. The overall importance of rainfall of each time step to discharge prediction (Equation 11) is computed for each site using all the samples. These importance scores are then used to infer the hydrological processes of the system being modeled. The physical realism of the inferred processes is evaluated using principles derived from hydrological knowledge of the system being modeled, which includes (a) smaller catchments commonly have faster responses to rainfalls compared to larger catchments and (b) the importance scores of rainfalls change smoothly across time steps. Principle (b) is derived from hydrological knowledge that rainfalls of similar magnitudes in adjacent time steps are expected to have similar impacts on the runoff generation processes.
3. This experiment aims to investigate whether different ML explanation methods lead to similar inferred hydrological processes. The gain, cover, and frequency metrics are computed for each XGBoost model of WS. These scores are then distributed among rainfall of each time step proportionally to its associated rainfall depth of all the samples, and the resulting quantities are compared to the normalized importance scores derived in experiment #2. The SHAP-related importance scores are normalized such that the resulting scores associated with all predictors sum to 1.
4. This experiment aims to investigate whether more accurate models are more likely to provide physically plausible explanations to the physical processes being modeled. The XGBoost models trained using 10%, 20%, 40%, and 60% of the observed samples of WS are evaluated in terms of their prediction accuracy and ability to provide consistent explanations to the modeled processes. The models' optimal hyperparameters are estimated using a resampling method (i.e., a training-validation split) and Bayesian optimization methods. The test dataset used in prediction accuracy estimation contains 40% of the samples and is the same for all models (that are trained on datasets of various sizes). The remaining 60% of the samples are then used for creating training datasets that contain 10%, 20%, 40%, and 60% of the samples. Note that the training dataset that contains 60% of the samples is created using all the remaining samples (that are not contained in the test dataset). For each training dataset, a further training-validation split is defined, and 10 models are then trained by repeatedly applying the Bayesian optimization methods (which are stochastic). For each sample size, the training dataset creation and training-validation splitting procedures are repeated 10 times that each produces 10 models (by applying Bayesian optimization methods repeatedly). That is, 100 versions of models are derived for each sample size. The importance score of rainfall of each time step is then derived following the methods in experiment #2, which is then used to infer the hydrological processes being modeled. The consistency of the inferred processes is then evaluated based on whether the importance scores of rainfalls change smoothly across time steps using the test dataset, where a threshold of 5e-5 L/s is used to account for numerical error.

3 Results and discussion

3.1 Prediction accuracy of machine learning models

The prediction accuracies of the various WS and SHC models are compared in Figure 5. The root-mean-square error (RMSE), coefficient of determination (R^2), and Nash-Sutcliffe coefficient of efficiency (NSE; Nash and Sutcliffe, 1970) of the predictions on the test datasets are compared, except for the SWMM model developed by Lee et al. (2018a), which was tested on a part of its training dataset due to small dataset size. The prediction accuracies of the XGBoost models, i.e., NSE > 0.7 and $R^2 > 0.7$, can be considered satisfactory, considering that they were relatively easy to set up and that it was impossible or very difficult to build process-based models for either site. The XGBoost models for both sites consistently outperform the linear regression (LM) models, suggesting that more sophisticated ML algorithms such as XGBoost are able to better capture the complex rainfall-runoff correlations than simple LM methods. The SHC XGBoost models have comparable prediction accuracies to SWMM, although the former were built with considerably less efforts. Thus, in future SuDS studies, it can be useful to quickly train some ML models based on available data and used them as a reference to evaluate the prediction accuracies of process-based models. The proposed ML model training methods can be potentially extended to study other small-scale urban catchments that have similar configurations to SHC.

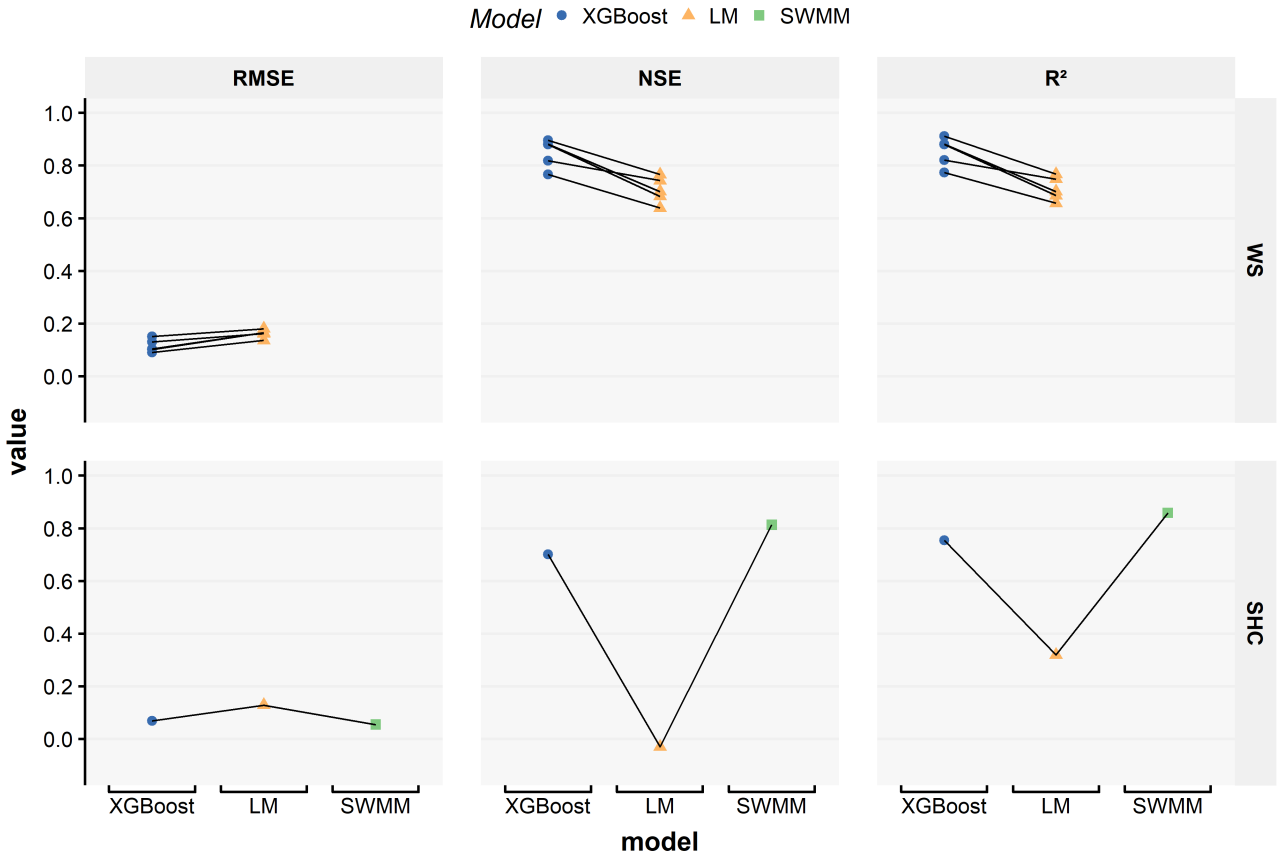

Figure 5 Prediction accuracies of the various models built for the Washington Street SuDS site (WS) and Shayler Crossing Watershed (SHC). The prediction accuracies are evaluated in terms of the root-mean-square error (RMSE), coefficient of determination (R^2), and Nash-Sutcliffe coefficient of efficiency (NSE). The RMSE units are L/s for WS and m^3/s for SHC. Each data point in the figure shows the prediction accuracy evaluated using a specific split of the dataset. The prediction accuracies derived using the same test dataset are connected by lines. The SWMM model for SHC was built by Lee et al. (2018a).

Each data point in Figure 5 shows the results obtained for a specific split of a dataset (i.e., the division of data into training, validation, and test datasets), and the points that correspond to the same split are connected by lines. The prediction accuracies of XGBoost and LM models varied considerably for different splits of a dataset. The variations indicate that the sample distributions in the different versions of training and test datasets appreciably differ, even though a stratified sampling method was used to balance the sample distribution in the different folds. The imbalanced sample distribution is associated with the limited number of samples used for the model training and evaluation, which implies that the four years of rainfall-runoff data of WS still contained an insufficient number of samples for the ML methods examined in this study. For instance, only a few high-flow events were observed each year in WS, which may be insufficient for training ML models that provide accurate

high-flow predictions. Even fewer samples were available for training and testing the SHC models; thus, the uncertainties of the prediction accuracies may be even larger.

3.2 Physical realism of the decomposed hydrographs

The results obtained for numerical experiment #1 are presented in this section. As an example, Figure 6 shows the predicted decomposed hydrographs of WS associated with rainfalls of different periods for a large, medium, and small runoff event. As shown in Figure 6, runoff is mostly contributed by the rainfall that occurred within the past 1 h regardless of the runoff event magnitude, especially for the peak discharge. These patterns are generally expected from small catchments, where runoffs are mostly contributed by recent rainfalls. As WS is a small-scale catchment, the inferred fast runoff responses are consistent with our hydrological knowledge.

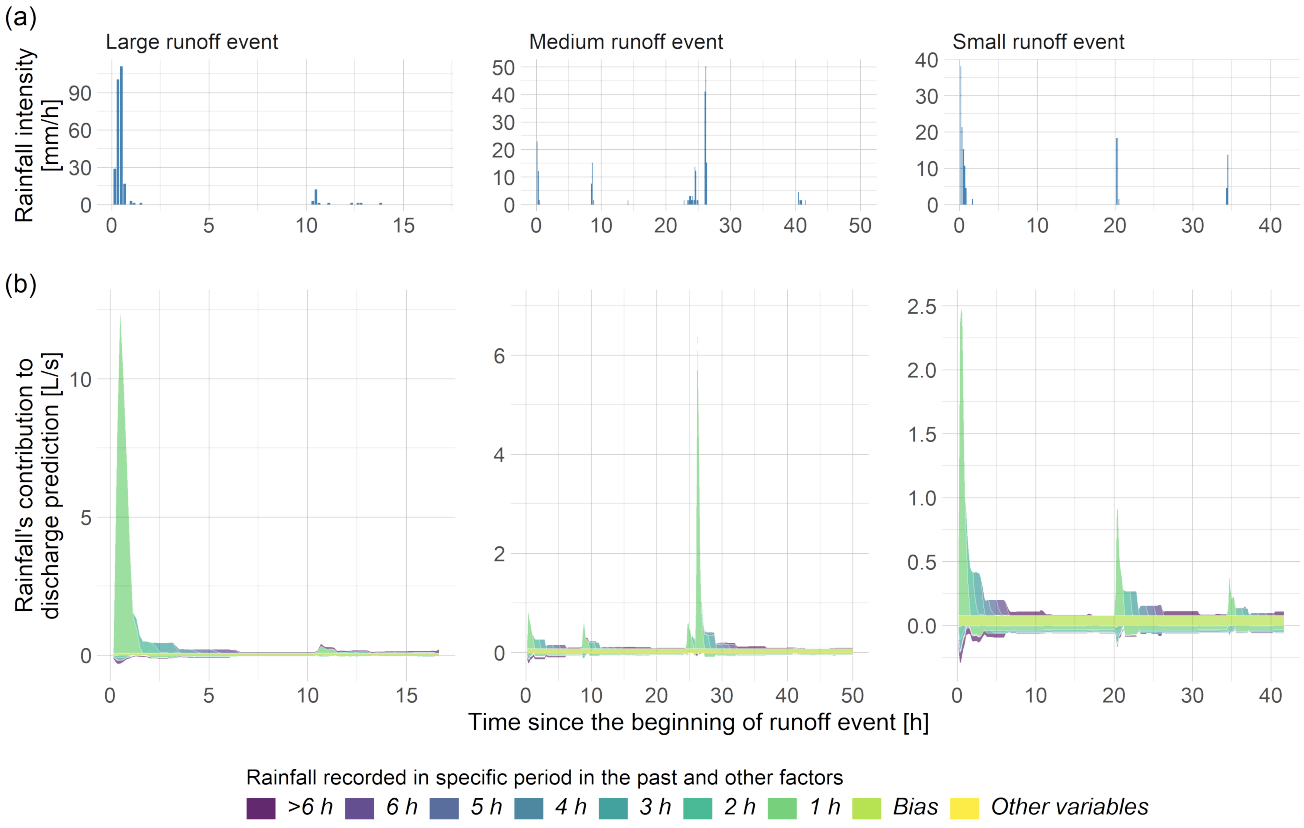

Figure 6 (a) Rainfall time series and (b) decomposed hydrographs of a large, medium, and small runoff event from the Washington Street SuDS site (WS). The model used to derive the hydrographs was obtained in the outer cross-validation iteration 1. Observational SHAP values were used in the computation.

Although it is not exactly clear how to quantitatively assess the physical realism of the contribution values assigned to rainfall of each hour, they express some patterns that are obviously inconsistent with our hydrological knowledge. First, Figure

6 shows that rainfalls can have negative contributions to runoffs, which are physically impossible. Although rainfall is the only forcing variable that is used by the model, the negative contributions may also be explained by the other "implicit" variables that can be predicted by rainfall, e.g., evapotranspiration is correlated with rainfall and can cause water losses from the catchment. However, it is nontrivial to exclude the contributions of the implicit variables such that rainfalls' contributions to runoffs are strictly nonnegative. Second, there is a constant bias term in the decomposed hydrographs that are independent of the rainfalls. This term is the average prediction for all samples and accounts for the differences between the predicted values and the contributions assigned to all variables in the SHAP method. However, this constant term does not clearly correspond to any processes in hydrology. Additionally, a model might use features that are not derived from rainfall (e.g., the age of the SuDS practice) as a predictor, which will also be assigned with contributions to runoffs when the SHAP method is used to examine the basis of predictions. However, it is unclear how to use these contributions in hydrograph separation.

The results of this experiment show that the inferred hydrological processes can be only partially consistent with the knowledge of the system being modeled. Some ML explanation methods, such as SHAP, can generate explanations that are inherently inconsistent with hydrological principles, such as the rainfalls' negative contributions to runoffs and the constant contributions to runoffs that are not associated with any variable. Nevertheless, it would be meaningful to compare the decomposed hydrographs to those derived using approaches in process-based modeling and tracer and isotope hydrology to further evaluate the validity of the explanations derived using ML methods.

3.3 Physical realism of the importance of rainfalls of different time steps to discharge prediction

The results of numerical experiment #2 are shown in Figure 7. In both WS and SHC, rainfalls recorded prior to 100 time steps in the past (i.e., 16.7 h) have almost no impact on discharge prediction, which is reasonable considering their small catchment sizes. The rainfalls that occurred in 1 and 4 time steps (i.e., 10 and 40 min) in the past are found to have the highest impact on discharge prediction for WS and SHC, respectively. This pattern is expected as SHC is considerably (which is around 800 times) larger than WS, and thereby the time required for stormwater to travel through the catchment is also longer in SHC. Although the exact response time of both catchments is unknown, it is possible to use the knowledge regarding the relations between the response time of the two catchments to conduct an assessment. Utilizing relational patterns in assessing the consistency of multiple entities has been demonstrated in Yang and Chui (2021).

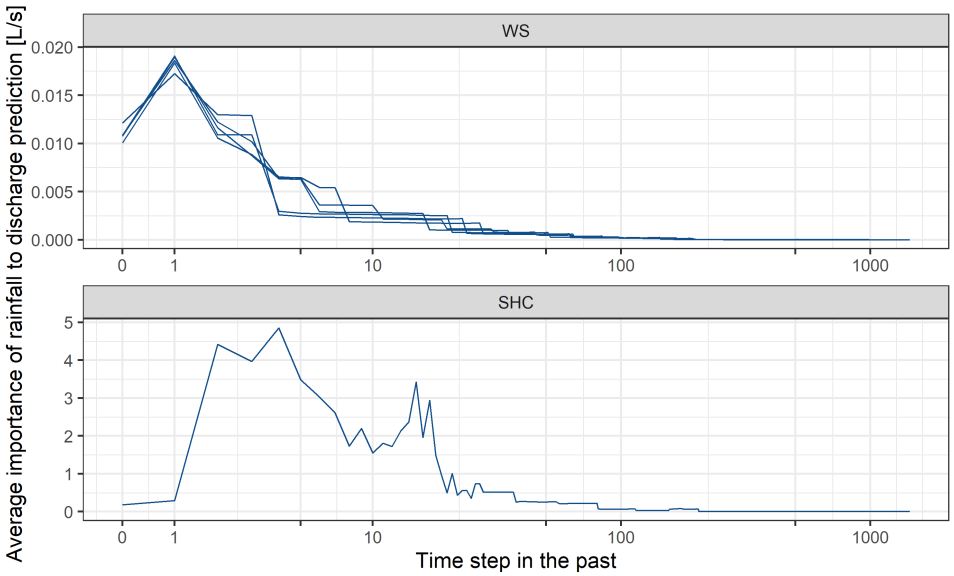

Figure 7 Average importance of the rainfalls at different time steps in the past for discharge predictions in the XGBoost models of the Washington Street SuDS site (WS) and the Shayler Crossing Watershed (SHC). Each line corresponds to an XGBoost model trained on a specific training dataset. Each time step is 10 min. The *x*-axis is on a pseudo-logarithmic scale. Observational SHAP values were used in the computation.

It is also worth noting that the importance scores assigned to rainfalls fluctuate across time steps for SHC, which indicates that the models find rainfalls in some specific time steps are more important for discharge prediction when compared to the others, which is inconsistent with our hydrological knowledge. In WS, these inconsistent patterns are not observed, where the importance scores of rainfalls generally change monotonically from the past 1 time step to the current time step and the time steps in the more distant past. The inconsistent patterns might be caused by the insufficient data used in model training and evaluation, which is discussed further in Section 3.5.

In WS, the rainfall of a specific time step can be assigned with notably different importance scores when the models trained using different training datasets are used, which is an indication of considerably different model structures. The structural differences in ML models are also reported in Schmidt et al. (2020), where the existence of multiple possible model structures is referred to as equifinality (Beven and Freer, 2001). The different importance scores will naturally result in different explanations of the processes being modeled, and apparently, these explanations cannot be simultaneously close to reality (Bouaziz et al., 2021). However, in this case, it is not possible to quantitively assess the explanations provided by different models due to the lack of knowledge. Experiment #4 investigate whether models with higher prediction accuracies can generate more trustworthy explanations to the process being modeled, and the results are presented in Section 3.5.

3.4 Comparison of multiple methods for explaining machine learning modeling results

The results of experiment #3 are shown in Figure 8, where various feature importance scores of rainfalls of different time steps are compared. All the importance scores derived using different methods suggest that the rainfalls of more recent time steps

have a higher impact on discharge prediction. However, different methods can derive considerably different importance scores for the rainfall of a specific time step, and the relations between the importance scores assigned to the rainfalls of different time steps also vary among different methods. Some methods, such as interventional SHAP and frequency, are more sensitive to model structural differences than the others. The differences in various importance scores indicate that the selection of the explanation method is another source of uncertainty in inferring the processes being modeled.

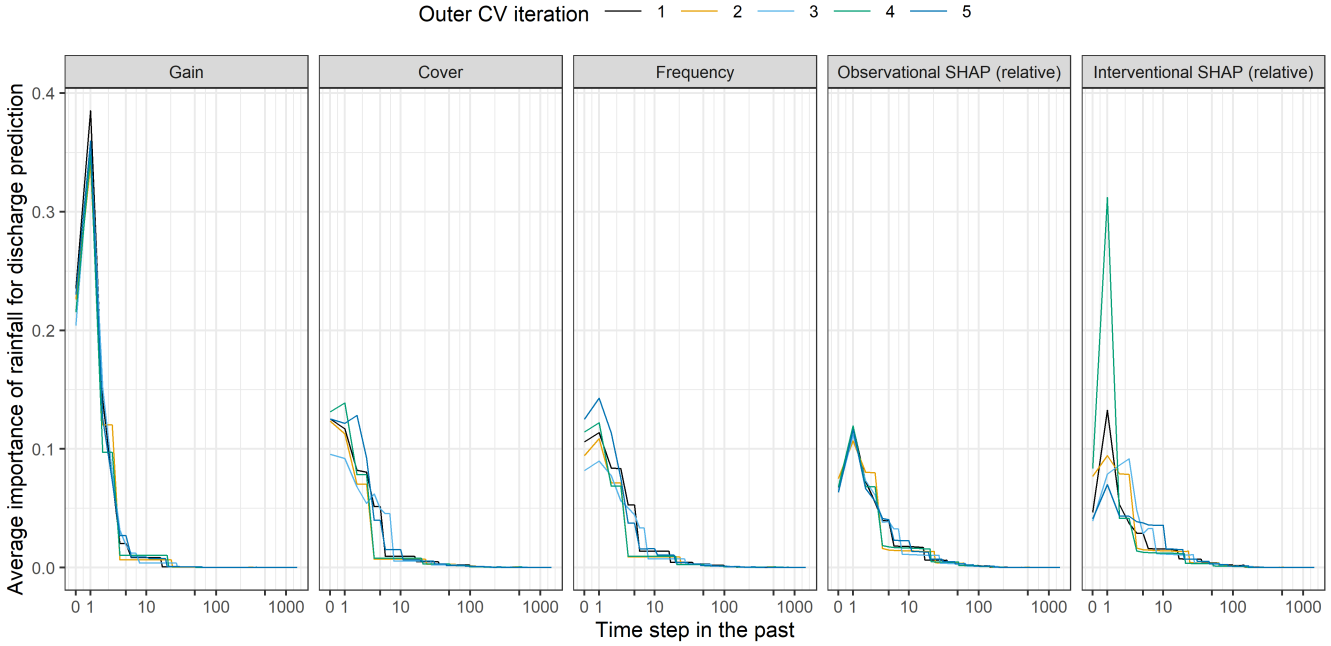

Figure 8 The importance of rainfall from each time step for making discharge predictions assessed by different feature importance measures in the Washington Street SuDS site (WS) XGBoost models. Each line shows the results of model trained during an outer cross-validation (CV) iteration. The *x*-axis is on a pseudo-logarithmic scale. For interventional SHAP, all the training samples are used as background dataset.

The importance scores derived from the observational and interventional SHAP values varied significantly due to the different methods used for computing the expected prediction, i.e., Equations 7 to 8. It is, however, currently unclear how to evaluate an explanation method's effectiveness in inferring the processes being modeled. Nevertheless, it is recommended that future study to always report the configurations of the explanation methods being used and evaluate the uncertainties associated with explanation method selection.

3.5 Correlation between the physical realism of inferred processes and model prediction accuracy

The results of experiment #4 are discussed in this section. As shown in Figure 9, a models' prediction accuracy, as measured by NSE, generally increases as more samples are used to train the model, despite its large uncertainties associated with the random sampling of the dataset. However, the number of models that correspond to consistent explanations (i.e., the single-peaked patterns of rainfall importance scores), also shown by the numbers in Figure 9, did not increase as more samples are

used in training. In fact, consistent results are not often observed for all training sample sizes. These results suggest that more accurate models do not necessarily offer more physically realistic explanations to the processes being modeled and using more data samples in model training does not guarantee more physically realistic explanations. For the models trained with the same amount of data, selecting a more accurate model does not guarantee that the inferences made based on the selected model are more consistent with our knowledge. However, it is also possible that our knowledge used in the assessment is biased, as the test dataset may not represent the true data distribution that would be observed on an infinite time scale.

Figure 9 Boxplot of the prediction accuracy, as measured by Nash-Sutcliffe coefficient of efficiency (NSE), of the Washington Street SuDS site (WS) XGBoost models when samples of different sizes are used in model training. For each sample size, the models that offer consistent or inconsistent explanations are grouped together, and the numbers of consistent and inconsistent models (the *n* values) are shown. Observational SHAP values were used in the computation.

Ilyas et al. (2019) argued that ML models can make predictions based on features that humans cannot comprehend. The implication of this argument is that ML models can make right predictions for the "wrong" reasons (Ross et al., 2017) or reasons that are inconsistent with our knowledge, and therefore the prediction accuracy of a model is not a trustworthy measurement of the physical realism of the explanations it provides. Regularizing ML models using physical principles, as suggested in Nearing et al. (2021), can potentially increase the physical realism of the explanations and the inferred physical processes.

3.6 Inferring hydrological processes using machine learning methods and analyzing ingredients of cake samples

This section uses a metaphor to explain the processes of inferring hydrological processes using ML methods, as shown in Figure 10. An ML model is similar to a cake (baked by others) in that they both are consumed by humans, and the exact mechanisms that generate the outcomes (i.e., the predictions or the tastes) are often unknown due to complexity. Here, the

mechanisms refer to the numerical operations that generate the predictions or the ingredients and procedures that give the flavor. Normally, the predictions or the tastes are of main interest. However, it can be useful to inspect the mechanisms that lead to the outcomes such that more confidence regarding future outcomes can be gained, where future outcomes refer to the predictions under new conditions or more cakes acquired through similar means. ML explanation methods and chemical composition analysis methods can provide information regarding the elements that contribute to the prediction or flavor. However, in many cases, such information can only be treated as circumstantial evidence of certain physical principles being learned by a model or the presence of certain ingredients. This is because ML models usually do not have structures that directly resemble the physical processes and chemical composition analysis usually does not directly test the presence of a food ingredient. The raw information is then further processed by referring to domain-specific knowledge, such as hydrological knowledge or nutritional facts of foods. For instance, a high degree of association between the predicted discharge and recent rainfall is an indication of a catchment's fast runoff response, which is commonly seen in small urban catchments, and a high carotene content may indicate that carrots are used in the cake. The inferred physical processes or ingredients and procedures are then evaluated against that would be expected based on domain-specific knowledge of the system being modeled or baking a specific type of cake. Finally, whether the ML model learns the expected physical processes or whether the cake is baked following the expected recipe is evaluated.

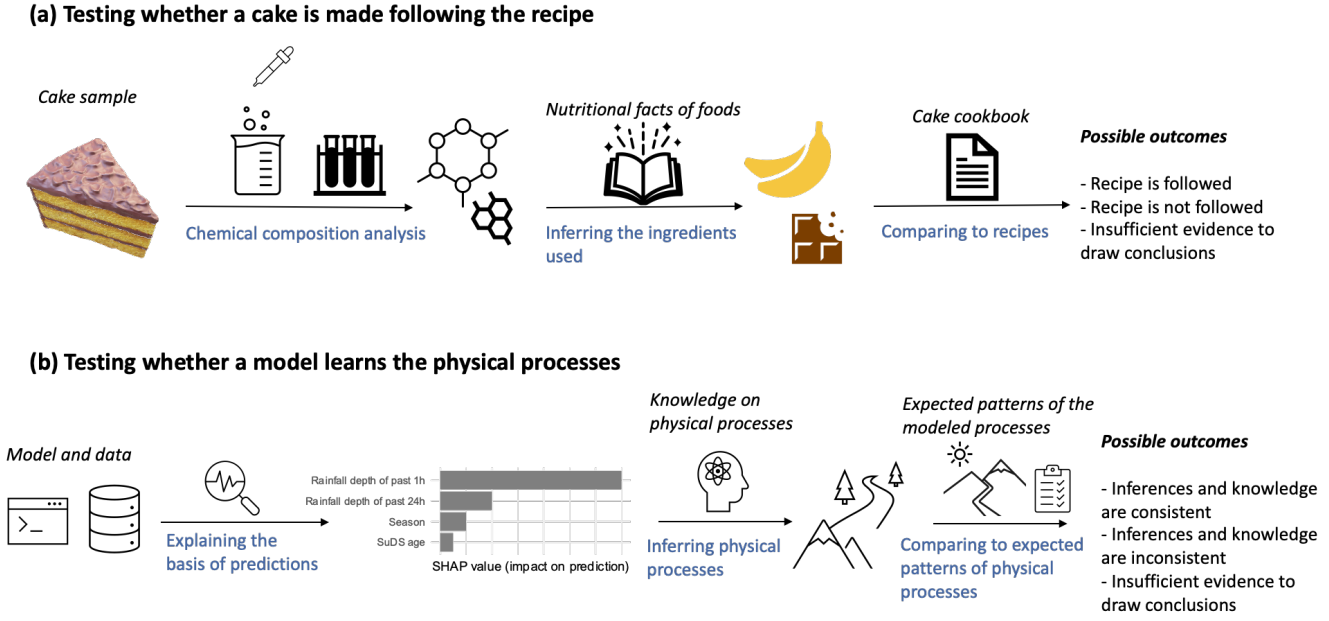

Figure 10 Illustration of the processes of (a) testing whether a cake is made following the desired recipe and (b) testing whether a model captures the physical processes of the system being modeled.

It is important to note that uncertainties are presented in every step of the assessment. First, different ML explanation methods or chemical composition analysis methods can lead to significantly different outcomes, which are used as the basis for inference. An example is provided in Section 3.4. Second, the inference process can be biased and subjective due to our

incomplete knowledge. For example, a chemical component could correspond to an ingredient that we do not know, and a rainfall's negative contribution to runoff could be caused by an unknown stormwater harvesting activity in the catchment. Bias and subjectivity, however, cannot be avoided in an open system, as there may be infinitely many physically plausible explanations to the same outcome (Oreskes et al., 1994). Third, the knowledge applied in the final assessment processes to define assessment criteria can be incomplete.

Are the practices of inferring the physical processes using ML methods any good? This study considers the consistency evaluation results of the inferences as circumstantial evidence to support a model's trustworthiness. Although an ML model does not have to learn the physical principles to make good predictions, we might prefer that the desired principles are captured by the models. This is similar to suggest that we prefer delicious cakes are made with ingredients we considered safe. Models that are associated with inferences that are consistent with our knowledge may be proven more reliable under new circumstances, such as extreme event predictions and predictions under data distribution drifts (Lu et al., 2019). More research on testing ML models' reliability is recommended.

It is also important to note that the inferred processes should be interpreted cautiously due to the large uncertainties involved in every step of the assessment. The detailed configurations of the entire inference process should be reported when presenting the inferred processes. In particular, large uncertainty resides in the process of making inferences according to the raw explanations derived from ML explanation methods, as many physical processes can give rise to the same raw explanations. It is therefore important to consider a larger search space for drawing inferences, which may be considered as an attempt to mitigate the streetlight effect, i.e., limiting the search space to be only under a streetlight in the dark or a specific set of plausible explanations (Demirdjian et al., 2005).

4 Conclusions

The following conclusions can be drawn.

1. This study shows that ML methods can be useful for modeling the hydrological responses of SuDS catchments at sub-hourly time scales. In this study, models with high prediction accuracies (NSE > 0.7) are obtained for two SuDS catchments of different sizes, SuDS practice types, and data availabilities. ML models can be set up relatively easily provided that observation data of the variables of interest are available and thus are recommended to be used as a reference to evaluate process-based models.
2. The physical process being modeled can be inferred based on the results of ML explanation methods. However, the inferred processes might be inconsistent with the patterns that would be expected based on domain-specific knowledge of the system being modeled. An ML model's ability to provide accurate predictions can be uncorrelated with its ability to offer plausible explanations to the physical processes being modeled.
3. This study provides a high-level overview of the processes of inferring the physical processes being modeled using ML explanation methods. It shows that large uncertainties are presented in the processes of explaining model

structures using ML explanation methods, making inferences according to the raw explanations, and assessing the physical realism of the inferred physical processes. The inferred hydrological processes normally should only be considered as circumstantial evidence to support a model's trustworthiness due to their indirect connection to the raw explanations. Due to the existence of the large uncertainties in the inference processes, the inferred physical processes should be interpreted cautiously, and more physically plausible explanations that correspond to the same raw explanations can be potentially investigated.

Code availability

The source code used in this study is available at https://github.com/stsfk/ExplainableML_SuDS, which contains functions for data pre-processing, modeling, modeling result analysis, and plotting. The following R packages are used for modeling and analysis in this research: xgboost (Chen and He, 2020), tidymodels (Kuhn and Wickham, 2020), lubridate (Grolemund and Wickham, 2011), RcppRoll (Ushey, 2018), zeallot (Teetor, 2018), mlrMBO (Bischl et al., 2017), and hydroGOF (Zambrano-Bigiarini, 2020). The following Python packages are used: shap (Lundberg and Lee, 2017), NumPy (Harris et al., 2020), and xgboost (Chen and Guestrin, 2016). All the R and Python packages used in this research are freely available online.

Data availability

The data of the two study sites examined in this study is obtained from the United States Geological Survey (USGS), Clermont County, Ohio, U.S., and the United States Environmental Protection Agency (U.S. EPA). The identification numbers of the USGS monitoring sites for the Washington Street SuDS site (WS) are 412533081221500, 412535081221400, and 412535081221402. The data of the Shayler Crossing Watershed can be downloaded at https://doi.org/10.23719/1378947. The SWMM model used in this research is developed in Lee et al. (2018a).

Author contribution

YY designed the study, acquired the data, wrote the code, conducted the numerical experiments, analyzed the results, and prepared and revised the manuscript. TFMC contributed to the design of numerical experiments, supervised the study, validated the results, and revised the manuscript.

Competing interests

The authors declare that they have no conflict of interest.

Acknowledgments

The work described in this paper was partly supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. HKU17255516), and partly supported by the RGC Theme-based Research Scheme (Grant No: T21-711/16-R) funded by the Research Grants Council of the Hong Kong Special Administrative Region, China. We thank Robert Darner from USGS, Christopher Nietch and Michelle Simon from U.S. EPA, Joong Gwang Lee from Center for Urban Green Infrastructure Engineering, and Bill Mellman from Clermont County Water Resources for providing data for this research. We thank the comments from Omar Wani and Hoshin Gupta on evaluating the physical realism of machine learning models in hydrological studies. We also would like to thank Georgia Papacharalampous and the other two anonymous reviewers for providing insightful feedback that greatly improved the quality of this paper. Finally, we wish to thank the editor Roberto Greco for handling and assessing our paper during the review processes.

References

Ahmad, M. A., Teredesai, A. and Eckert, C.: Interpretable machine learning in healthcare, in Proceedings - 2018 IEEE International Conference on Healthcare Informatics, ICHI 2018, p. 447., 2018.

Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R. and Herrera, F.: Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI, Inf. Fusion, 58, 82–115, doi:10.1016/J.INFFUS.2019.12.012, 2020.

Beven, K. and Freer, J.: Equifinality, data assimilation, and uncertainty estimation in mechanistic modelling of complex environmental systems using the GLUE methodology, J. Hydrol., 249(1–4), 11–29, doi:10.1016/S0022-1694(01)00421-8, 2001.

Bischl, B., Richter, J., Bossek, J., Horn, D., Thomas, J. and Lang, M.: mlrMBO: A Modular Framework for Model-Based Optimization of Expensive Black-Box Functions, [online] Available from: https://arxiv.org/abs/1703.03373v3 (Accessed 4 September 2021), 2017.

Bojanowski, P., Joulin, A., Paz, D. L. and Szlam, A.: Optimizing the latent space of generative networks, in 35th International Conference on Machine Learning, ICML 2018, vol. 2, pp. 960–972., 2018.

Bouaziz, L. J. E., Fenicia, F., Thirel, G., De Boer-Euser, T., Buitink, J., Brauer, C. C., De Niel, J., Dewals, B. J., Drogue, G., Grelier, B., Melsen, L. A., Moustakas, S., Nossent, J., Pereira, F., Sprokkereef, E., Stam, J., Weerts, A. H., Willems, P., Savenije, H. H. G. and Hrachowitz, M.: Behind the scenes of streamflow model performance, Hydrol. Earth Syst. Sci., 25(2), 1069–1095, doi:10.5194/hess-25-1069-2021, 2021.

Charlesworth, S. M.: A review of the adaptation and mitigation of global climate change using sustainable drainage in cities, J. Water Clim. Chang., 1(3), 165–180, doi:10.2166/wcc.2010.035, 2010.

Chen, H., Janizek, J. D., Lundberg, S. and Lee, S. I.: True to the model or true to the data?, arXiv [online] Available from: http://arxiv.org/abs/1805.11783, 2020.

Chen, T. and Guestrin, C.: XGBoost: A scalable tree boosting system, in Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, vol. 13-17-Augu, pp. 785–794., 2016.

Chen, T. and He, T.: xgboost: eXtreme Gradient Boosting, [online] Available from: https://cran.r-project.org/web/packages/xgboost/vignettes/xgboost.pdf (Accessed 29 June 2020), 2020.

Damodaram, C., Giacomoni, M. H., Prakash Khedun, C., Holmes, H., Ryan, A., Saour, W. and Zechman, E. M.: Simulation of combined best management practices and low impact development for sustainable stormwater management, J. Am. Water Resour. Assoc., 46(5), 907–918, doi:10.1111/j.1752-1688.2010.00462.x, 2010.

Darner, R. A. and Dumouchelle, D. H.: Hydraulic Characteristics of Low-Impact Development Practices in Northeastern Ohio, 2008-2010: U.S. Geological Survey Scientific Investigations Report 2011–5165. [online] Available from: https://pubs.usgs.gov/sir/2011/5165/ (Accessed 7 July 2020), 2011.

Darner, R. A., Shuster, W. D. and Dumouchelle, D. H.: Hydrologic Characteristics of Low-Impact Stormwater Control Measures at Two Sites in Northeastern Ohio , 2008 – 13: U.S. Geological Survey Scientific Investigations Report 2015-5030., 2015.

DeBusk, K. M., Hunt, W. F. and Line, D. E.: Bioretention Outflow: Does It Mimic Nonurban Watershed Shallow Interflow?, J. Hydrol. Eng., 16(3), 274–279, doi:10.1061/(ASCE)HE.1943-5584.0000315, 2011.

Demirdjian, D., Taycher, L., Shakhnarovich, G., Grauman, K. and Darrell, T.: Avoiding the “streetlight effect”: Tracking by exploring likelihood modes, in Proceedings of the IEEE International Conference on Computer Vision, vol. I, pp. 357–364., 2005.

Eckart, K., McPhee, Z. and Bolisetti, T.: Performance and implementation of low impact development – A review, Sci. Total Environ., 607–608, 413–432, doi:10.1016/j.scitotenv.2017.06.254, 2017.

Elliott, A. H. and Trowsdale, S. A.: A review of models for low impact urban stormwater drainage, Environ. Model. Softw., 22(3), 394–405, doi:10.1016/j.envsoft.2005.12.005, 2007.

Eric, M., Li, J. and Joksimovic, D.: Performance Evaluation of Low Impact Development Practices Using Linear Regression, Br. J. Environ. Clim. Chang., 5(2), 78–90, doi:10.9734/bjecc/2015/11578, 2015.

Fassman-Beck, E., Hunt, W., Berghage, R., Carpenter, D., Kurtz, T., Stovin, V. and Wadzuk, B.: Curve number and runoff coefficients for extensive living roofs, J. Hydrol. Eng., 21(3), 04015073, doi:10.1061/(ASCE)HE.1943-5584.0001318, 2016.

Fletcher, T. D., Shuster, W., Hunt, W. F., Ashley, R., Butler, D., Arthur, S., Trowsdale, S., Barraud, S., Semadeni-Davies, A., Bertrand-Krajewski, J. L., Mikkelsen, P. S., Rivard, G., Uhl, M., Dagenais, D. and Viklander, M.: SUDS, LID, BMPs, WSUD and more – The evolution and application of terminology surrounding urban drainage, Urban Water J., 12(7), 525–542, doi:10.1080/1573062X.2014.916314, 2015.

Frazier, P. I.: A tutorial on bayesian optimization, arXiv [online] Available from: http://arxiv.org/abs/1807.02811 (Accessed 6 October 2020), 2018.

Friedman, J. H.: Greedy function approximation: A gradient boosting machine, Ann. Stat., 29(5), 1189–1232, doi:10.1214/aos/1013203451, 2001.

Gimenez-Maranges, M., Breuste, J. and Hof, A.: Sustainable Drainage Systems for transitioning to sustainable urban flood management in the European Union: A review, J. Clean. Prod., 255, 120191, doi:10.1016/j.jclepro.2020.120191, 2020.

Grolemund, G, and Wickham, H.: Dates and Times Made Easy with lubridate. Journal of Statistical Software, 40(3), 1-25, https://www.jstatsoft.org/v40/i03/, 2011.

Gauch, M., Kratzert, F., Klotz, D., Nearing, G., Lin, J., and Hochreiter, S.: Rainfall-runoff prediction at multiple timescales with a single Long Short-Term Memory network, Hydrol. Earth Syst. Sci., 25, 2045–2062, https://doi.org/10.5194/hess-25-2045-2021, 2021.

Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F. and Pedreschi, D.: A Survey of Methods for Explaining Black Box Models, ACM Comput. Surv., 51(5), 1–42, doi:10.1145/3236009, 2019.

Guo, Y. and Senior, M. J.: Climate model simulation of point rainfall frequency characteristics, J. Hydrol. Eng., 11(6), 547–554, doi:10.1061/(ASCE)1084-0699(2006)11:6(547), 2006.

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., Gérard-Marchant, P., Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C. and Oliphant, T. E.: Array programming with NumPy, Nature, 585(7825), 357–362, doi:10.1038/s41586-020-2649-2, 2020.

Hastie, T., Tibshirani, R. and Friedman, J.: The Elements of Statistical Learning, Springer New York, New York, NY., 2009.

Hoghooghi, N., Golden, H. E., Bledsoe, B. P., Barnhart, B. L., Brookes, A. F., Djang, K. S., Halama, J. J., McKane, R. B., Nietch, C. T. and Pettus, P. P.: Cumulative effects of Low Impact Development on watershed hydrology in a mixed land-cover system, Water (Switzerland), 10(8), 991, doi:10.3390/w10080991, 2018.

Hopkins, K. G., Bhaskar, A. S., Woznicki, S. A. and Fanelli, R. M.: Changes in event-based streamflow magnitude and timing after suburban development with infiltration-based stormwater management, Hydrol. Process., 34(2), 387–403, doi:10.1002/hyp.13593, 2020.

Ilyas, A., Santurkar, S., Tsipras, D., Engstrom, L., Tran, B. and Madry, A.: Adversarial examples are not bugs, they are features, in Advances in Neural Information Processing Systems, vol. 32. [online] Available from: http://git.io/adv-datasets. (Accessed 28 June 2021), 2019.

Janzing, D., Minorics, L. and Blöbaum, P.: Feature relevance quantification in explainable ai: A causal problem, arXiv, 2019.

Johannessen, B. G., Hanslin, H. M. and Muthanna, T. M.: Green roof performance potential in cold and wet regions, Ecol. Eng., 106, 436–447, doi:10.1016/j.ecoleng.2017.06.011, 2017.

Jones, P. and Macdonald, N.: Making space for unruly water: Sustainable drainage systems and the disciplining of surface runoff, Geoforum, 38(3), 534–544, doi:10.1016/j.geoforum.2006.10.005, 2007.

Khan, U. T., Valeo, C., Chu, A. and He, J.: A data driven approach to bioretention cell performance: Prediction and design, Water (Switzerland), 5(1), 13–28, doi:10.3390/w5010013, 2013.

Kirchner, J. W.: Getting the right answers for the right reasons: Linking measurements, analyses, and models to advance the science of hydrology, Water Resour. Res., 42(3), doi:10.1029/2005WR004362, 2006.

Kratzert, F., Herrnegger, M., Klotz, D., Hochreiter, S. and Klambauer, G.: NeuralHydrology – Interpreting LSTMs in Hydrology, in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 11700 LNCS, pp. 347–362., 2019.

Kuhn, M. and Johnson, K.: Applied predictive modeling., 2013.

Kuhn, M. and Johnson, K.: Feature Engineering and Selection : a Practical Approach for Predictive Models., Chapman and Hall/CRC. [online] Available from: https://www.routledge.com/Feature-Engineering-and-Selection-A-Practical-Approach-for-Predictive-Models/Kuhn-Johnson/p/book/9781138079229 (Accessed 24 July 2020), 2019.

Kuhn, M. and Wickham, H.: Tidymodels: a collection of packages for modeling and machine learning using tidyverse principles, https://www.tidymodels.org, 2020.

Lee, J. G., Nietch, C. T. and Panguluri, S.: Drainage area characterization for evaluating green infrastructure using the Storm Water Management Model, Hydrol. Earth Syst. Sci., 22(5), 2615–2635, doi:10.5194/hess-22-2615-2018, 2018a.

Lee, J. G., Nietch, C. T. and Panguluri, S.: SWMM Modeling Methods for Simulating Green Infrastructure at a Suburban Headwatershed: User's Guide, U.S. Environ. Prot. Agency, (October), 157 [online] Available from: https://nepis.epa.gov/Exe/ZyPDF.cgi/P100TJ39.PDF?Dockey=P100TJ39.PDF%0A (Accessed 11 July 2020b), 2018.

Li, S., Kazemi, H. and Rockaway, T. D.: Performance assessment of stormwater GI practices using artificial neural networks, Sci. Total Environ., 651, 2811–2819, doi:10.1016/j.scitotenv.2018.10.155, 2019.

Liu, J., Sample, D., Bell, C. and Guan, Y.: Review and Research Needs of Bioretention Used for the Treatment of Urban Stormwater, Water, 6(4), 1069–1099, doi:10.3390/w6041069, 2014.

Lu, J., Liu, A., Dong, F., Gu, F., Gama, J. and Zhang, G.: Learning under Concept Drift: A Review, IEEE Trans. Knowl. Data Eng., 31(12), 2346–2363, doi:10.1109/TKDE.2018.2876857, 2019.

Lundberg, S. M. and Lee, S. I.: A unified approach to interpreting model predictions, in Advances in Neural Information Processing Systems, vol. 2017-Decem, pp. 4766–4775. [online] Available from: https://github.com/slundberg/shap (Accessed 30 June 2020), 2017.

Lundberg, S. M., Erion, G., Chen, H., DeGrave, A., Prutkin, J. M., Nair, B., Katz, R., Himmelfarb, J., Bansal, N. and Lee, S.-I.: From local explanations to global understanding with explainable AI for trees, Nat. Mach. Intell., 2(1), 56–67, doi:10.1038/s42256-019-0138-9, 2020.

Maier, H. R. and Dandy, G. C.: Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications, Environ. Model. Softw., 15(1), 101–124, doi:10.1016/S1364-8152(99)00007-9, 2000.

Mitchell, R. and Frank, E.: Accelerating the XGBoost algorithm using GPU computing, PeerJ Comput. Sci., 2017(7), e127, doi:10.7717/peerj-cs.127, 2017.

Montalto, F., Behr, C., Alfredo, K., Wolf, M., Arye, M. and Walsh, M.: Rapid assessment of the cost-effectiveness of low impact development for CSO control, Landsc. Urban Plan., 82(3), 117–131, doi:10.1016/j.landurbplan.2007.02.004, 2007.

Morton, A.: Mathematical models: Questions of trustworthiness, Br. J. Philos. Sci., 44(4), 659–674, doi:10.1093/bjps/44.4.659, 1993.

Muthanna, T. M., Viklander, M. and Thorolfsson, S. T.: Seasonal climatic effects on the hydrology of a rain garden, Hydrol. Process., 22(11), 1640–1649, doi:10.1002/hyp.6732, 2008.

Muthoo, A., Osborne, M. J. and Rubinstein, A.: A Course in Game Theory., Economica, 63(249), 164, doi:10.2307/2554642, 1996.

Nash, J. E. and Sutcliffe, J. V.: River flow forecasting through conceptual models part I - A discussion of principles, J. Hydrol., 10(3), 282–290, doi:10.1016/0022-1694(70)90255-6, 1970.

Nearing, G. S., Kratzert, F., Sampson, A. K., Pelissier, C. S., Klotz, D., Frame, J. M., Prieto, C. and Gupta, H. V: What Role Does Hydrological Science Play in the Age of Machine Learning?, Water Resour. Res., 57(3), doi:10.1029/2020WR028091, 2021.

Niazi, M., Nietch, C., Maghrebi, M., Jackson, N., Bennett, B. R., Tryby, M. and Massoudieh, A.: Storm Water Management Model: Performance Review and Gap Analysis, J. Sustain. Water Built Environ., 3(2), 04017002, doi:10.1061/JSWBAY.0000817, 2017.

Nielsen, A.: Practical Time Series Analysis, O'Reilly Media, Inc. [online] Available from: https://www.oreilly.com/library/view/practical-time-series/9781492041641/ (Accessed 30 June 2020), 2019.

Nielsen, D.: Tree Boosting With XGBoost: Why does XGBoost win every machine learning competition?, Master's Thesis, Norweigian Univ. Sci. Technol., (December), 2016, doi:10.1111/j.1758-5899.2011.00096.x, 2016.

Oreskes, N., Shrader-Frechette, K. and Belitz, K.: Verification, validation, and confirmation of numerical models in the earth sciences, Science (80-.)., 263(5147), 641–646, doi:10.1126/science.263.5147.641, 1994.

Pelletier, A. and Andréassian, V.: Hydrograph separation: An impartial parametrisation for an imperfect method, Hydrol. Earth Syst. Sci., 24(3), 1171–1187, doi:10.5194/hess-24-1171-2020, 2020.

Rosa, D. J., Clausen, J. C. and Dietz, M. E.: Calibration and Verification of SWMM for Low Impact Development, J. Am. Water Resour. Assoc., 1–12, doi:10.1111/jawr, 2015.

Ross, A., Hughes, M. C. and Doshi-Velez, F.: Right for the Right Reasons: Training Differentiable Models by Constraining their Explanations, [online] Available from: https://github.com/dtak/rrr. (Accessed 2 September 2021), 2017.

Rudin, C.: Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead, Nat. Mach. Intell., 1(5), 206–215, doi:10.1038/s42256-019-0048-x, 2019.

Schmidt, L., Heße, F., Attinger, S. and Kumar, R.: Challenges in Applying Machine Learning Models for Hydrological Inference: A Case Study for Flooding Events Across Germany, Water Resour. Res., 56(5), doi:10.1029/2019WR025924, 2020.

Selbig, W. R., Buer, N. and Danz, M. E.: Stormwater-quality performance of lined permeable pavement systems, J. Environ. Manage., 251, doi:10.1016/j.jenvman.2019.109510, 2019.

Shahriari, B., Swersky, K., Wang, Z., Adams, R. P. and De Freitas, N.: Taking the human out of the loop: A review of Bayesian optimization, Proc. IEEE, 104(1), 148–175, doi:10.1109/JPROC.2015.2494218, 2016.

Snoek, J., Larochelle, H. and Adams, R. P.: Practical Bayesian optimization of machine learning algorithms, in Advances in Neural Information Processing Systems, vol. 4, pp. 2951–2959. [online] Available from: https://arxiv.org/abs/1206.2944v2

(Accessed 5 October 2020), 2012.

Shapley, L. S. A value of n-person games. Contributions to the Theory of Games, pp. 307–317, 1953.

Solomatine, D. P. and Dulal, K. N.: Model trees as an alternative to neural networks in rainfall-runoff modelling, Hydrol. Sci. J., 48(3), 399–411, doi:10.1623/hysj.48.3.399.45291, 2003.

Solomatine, D. P. and Ostfeld, A.: Data-driven modelling: Some past experiences and new approaches, in Journal of Hydroinformatics, vol. 10, pp. 3–22, IWA Publishing., 2008.

Starn, J. J., Kauffman, L. J., Carlson, C. S., Reddy, J. E. and Fienen, M. N.: Three-Dimensional Distribution of Groundwater Residence Time Metrics in the Glaciated United States Using Metamodels Trained on General Numerical Simulation Models, Water Resour. Res., 57(2), e2020WR027335, doi:10.1029/2020WR027335, 2021.

Sundararajan, M. and Najmi, A.: The many shapley values for model explanation, in 37th International Conference on Machine Learning, ICML 2020, vol. PartF16814, pp. 9210–9220., 2020.

Sundararajan, M., Taly, A. and Yan, Q.: Axiomatic attribution for deep networks, in 34th International Conference on Machine Learning, ICML 2017, vol. 7, pp. 5109–5118., 2017.

Teetor, N.: zeallot: Multiple, Unpacking, and Destructuring Assignment. R package version 0.1.0. https://CRAN.R-project.org/package=zeallot, 2018.

Trinh, D. H. and Chui, T. F. M.: Assessing the hydrologic restoration of an urbanized area via an integrated distributed hydrological model, Hydrol. Earth Syst. Sci., 17(12), 4789–4801, doi:10.5194/hess-17-4789-2013, 2013.

Ushey, K.: RcppRoll: Efficient Rolling / Windowed Operations. R package version 0.3.0. https://CRAN.R-project.org/package=RcppRoll, 2018.

Wani, O., Beckers, J. V. L., Weerts, A. H. and Solomatine, D. P.: Residual uncertainty estimation using instance-based learning with applications to hydrologic forecasting, Hydrol. Earth Syst. Sci., 21(8), 4021–4036, doi:10.5194/hess-21-4021-2017, 2017.

Yang, Y. and Chui, T. F. M.: Hydrologic Performance Simulation of Green Infrastructures: Why Data-Driven Modelling Can Be Useful?, in New Trends in Urban Drainage Modelling, pp. 480–484, Springer International Publishing., 2019.

Yang, Y. and Chui, T. F. M.: Reliability Assessment of Machine Learning Models in Hydrological Predictions through Metamorphic Testing, Water Resour. Res., 1–27, doi:10.1029/2020wr029471, 2021.

Yong, C. F., McCarthy, D. T. and Deletic, A.: Predicting physical clogging of porous and permeable pavements, J. Hydrol., 481, 48–55, doi:10.1016/j.jhydrol.2012.12.009, 2013.

Zambrano-Bigiarini, M.: hydroGOF: Goodness-of-fit functions for comparison of simulated and observed hydrological time seriesR package version 0.4-0., https://github.com/hzambran/hydroGOF, http://doi.org/10.5281/zenodo.840087, 2020.

Zeng, X. and Martinez, T. R.: Distribution-balanced stratified cross-validation for accuracy estimation, J. Exp. Theor. Artif. Intell., 12(1), 1–12, doi:10.1080/095281300146272, 2000.

Zhang, K. and Chui, T. F. M.: A review on implementing infiltration-based green infrastructure in shallow groundwater environments: Challenges, approaches, and progress, J. Hydrol., 579, 124089, doi:10.1016/j.jhydrol.2019.124089, 2019.

Zhou, Q.: A Review of Sustainable Urban Drainage Systems Considering the Climate Change and Urbanization Impacts,

Water, 6(4), 976–992, doi:10.3390/w6040976, 2014.