

Physics-Guided Meta-Learning Method in Baseflow Prediction over Large Regions

Shengyu Chen¹, Yiqun Xie², Xiang Li³, Xu Liang¹, Xiaowei Jia¹

¹ University of Pittsburgh, ²University of Maryland, ³ University of Minnesota

¹{shc160,xuliang,xiaowei}@pitt.edu, ²xie@umd.edu, ³lix5000@umn.edu

Abstract

Physics-based groundwater flow equations are powerful tools for water resource assessment under different hydrological and climatic conditions. How these conditions affect the discharge of groundwater (i.e., baseflow) into rivers is one of the most important topics in the hydrology domain. However, due to the different environmental conditions in different basins, it is difficult to use a single physics-based equation to represent the discharge of groundwater in all river basins. Despite the promise of data-driven models in capturing complex relationships, they are also limited in learning heterogeneous baseflow patterns from multiple basins, especially with sparse training data. In this paper, we propose a new data-driven model Physics Guided MeTa Learning (PGMTL), which uses meta-learning to adapt the predictive model to multiple basins and also enhance the meta-learning process with knowledge embodied in different physics-based equations so as to improve the baseflow prediction over a large number of river basins. Experimental results show that our proposed PGMTL has a significant improvement over either physics-based equations or ML models. Moreover, our method has been shown to perform much better with sparse or localized training data. Finally, our method is able to interpret the contribution of each physics-based equation under different scenarios.

Keywords: Physics-guided Meta-learning, Baseflow prediction, Clustering.

1 Introduction

Baseflow, also referred to as low flow, is the portion of the stream flow that is sustained between rainfall events and also during dry periods. Baseflow is contributed by slowly moving water from the groundwater aquifer and the slow flow within the porous media due to soil moisture and other delayed sources. Baseflow is essential for ecosystem functioning and provides habitat for stream biota, and it affects the water quality. Improving the reliability of baseflow prediction is critically important to water resources management, especially with the

changing climate and continuing population growth. In addition, better quantification of the baseflow lays the scientific foundation for investigating groundwater and surface water interactions, analysis of baseflow yields, and estimation of groundwater storage change, which is not only important for catchment hydrology [1] but also for the regional and global water and energy balance and climate studies as it directly affects soil moisture and groundwater storage [2]. Furthermore, a better understanding of groundwater storage is critical for making better strategies in dealing with extended drought since groundwater is an important buffer against climate variability by providing a secure water supply.

At present, baseflow for a watershed is often predicted by using either empirical [3], semi-empirical [4], or theoretically-based expressions/parameterizations as a function of soil moisture, groundwater table depth, soil properties, and watershed characteristics [5, 6]; or by using comprehensive groundwater numerical models [7]. Tallaksen [8] provided a comprehensive overview of the various baseflow relationships/parameterizations. However, all of these approaches have their significant limitations. For example, the empirical and semi-empirical expressions used to represent the baseflow behaviors are usually obtained according to specific conditions and observations and thus are only applicable to specific situations. The physically-derived relationships have theories to support but suffer various assumptions and simplifications made in the derivations and are generally applicable to relatively homogeneous watersheds. Therefore, the current widely used baseflow relationships/expressions cannot represent well the nonlinear behaviors involved in the baseflow dynamics in all situations. Besides, there are various parameters and coefficients involved in these relationships that cannot be adequately estimated due to limited available information and the current understanding of the complex problems. Predicting the baseflow using the groundwater numerical model approach is not only computationally expensive but also has various uncertainties involved as well, such as uncertainties due to limitations of our under-

standing of various hydrological processes, parameterizations, and simplifications of these processes, model structures, a large number of model parameters associated with soil properties (e.g., hydraulic conductivities) that need to be estimated with limited data, etc.

Machine learning (ML) models, given their tremendous success in several commercial applications (e.g., computer vision, and natural language processing), are increasingly being considered promising alternatives to physics-based equations in the scientific domains. For example, the long short-term memory (LSTM) model [9] has been widely used in modeling temporal patterns of water dynamics in the hydrology domain [10, 11]. Prior works have also used graph neural networks to model the interactions amongst streams [12, 13]. In streamflow problems, Moshe et al. [12] proposed the HydroNets model, which uses the ML model to integrate the information from river segments and their upstream segments to improve the streamflow predictions. Furthermore, Chen et al. [13] proposed an HRGN model to represent underlying stream-reservoir networks and improve streamflow temperature prediction in all river segments within a river network.

However, there are still several challenges faced by existing methods. Firstly, scientific problems often exhibit complex patterns amongst multiple physical variables. Standard ML approaches are not designed to capture such complex patterns solely from data. Secondly, given the substantial labor and material cost of collecting data, labeled data samples (observations) are limited. Third, different river basins exhibit distinct baseflow behaviors due to the variation of hydrological conditions (e.g., soil property and land covers). It increases the difficulty of using ML models to capture underlying relationships collectively from all basins.

In this work, we develop a new data-driven method Physics Guided MeTa Learning (PGMTL) to improve the prediction of baseflow dynamics for a large number of basins. The central idea is to leverage physical knowledge in adapting the ML model to different basins. Specifically, we use hydrological conditions of basins to create multiple tasks through a clustering process. Then we incorporate additional physical theories when the global ML model is adapted to each task. To facilitate the learning of complex baseflow patterns, we propose three different methods to enhance model predictions using simulated baseflow data generated by multiple physics-based equations.

We evaluate the proposed method in the 60 different river basins in Pennsylvania. The results demonstrate the superiority of the proposed method under whatever conditions with sufficient data or sparse data. The experimental results also show the effectiveness of the pro-

posed meta-learning method in extracting underlying physical relationships from simulations.

2 Related Work

Observations are used to evaluate various approaches for baseflow simulations and predictions, such as the various baseflow expressions, groundwater numerical models, ML methods, and hybrid methods. However, baseflow is not as directly observable as streamflow is. Thus, baseflow (the slow component) is obtained by separating it from the surface flow (the fast component) based on measured streamflow data. Common approaches to separate the baseflow from streamflow include the chemical balance approach (CBM) [14], and hydrograph separation (HS) approach [15, 16, 17]. The HS approach is widely used in hydrology and it includes different methods, such as graphical method [18], fixed and sliding interval method [19], local minimum methods [19, 20], and digital filter methods [15, 16, 17]. Each method has its strengths and limitations but past studies showed that the choice of different digital filters and local minimum methods provide comparable results [21].

Recent research has shown immense success in integrating physics knowledge into ML models to improve predictive performance and solve general scientific problems. The most common ways include applying additional loss functions [22, 23] and other hybrid approaches [24, 22]. In a recent survey [25], it summarized existing literature and approaches for integrating scientific knowledge into machine learning models. For example, Hanson et al. [26] developed a PGRNN model to introduce ecological principles as physical constraints into the loss function of ML models to improve the lake surface water phosphorus prediction. Karpatne et al. [22] proposed a hybrid ML and physics model which can introduce physics models into an ML model as additional input. This model can guarantee the physical relationship that the density of water at a lower depth is always greater than the density of water at any depth above. Then, our previous works [11, 23] further proposed new methods to reduce search space and improve prediction accuracy by including an additional penalty term for violating the law of energy conservation.

Meta-learning leverages a variety of general knowledge across a set of learning tasks to learn how to learn effectively when given a new task. The optimization-based [27, 28] meta-learning methods can adjust the optimization algorithm directly and adapt to new task learning quickly. For example, the Model agnostic meta-learning (MAML) [28] method consists of learning an initial set of model weights that are optimized to efficiently learn new tasks. Furthermore, the meta-learning method has been widely used in solving scientific prob-

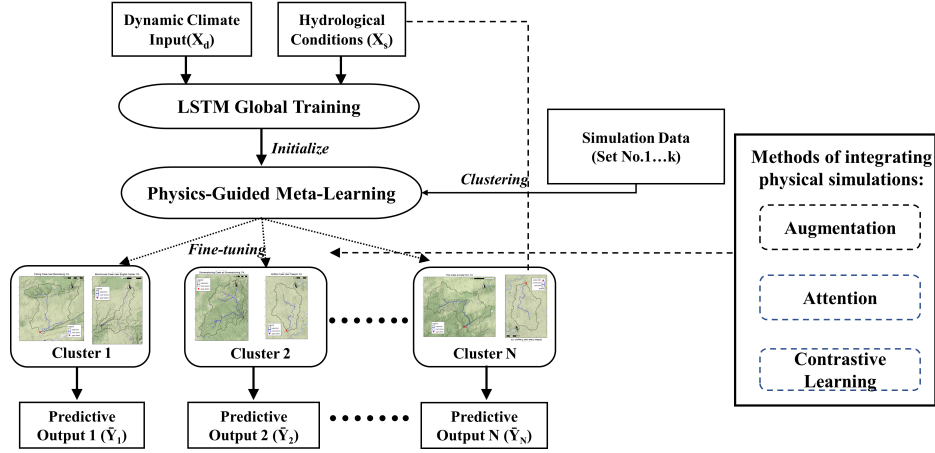


Figure 1: The overall structure of the proposed PGMTL framework.

lems [29, 30, 31]. For example, Liu et al. [30] integrated the meta-learning methods into the physics-informed neural networks (PINN) [32] to reduce the computation complexity and achieved higher accuracy in solving partial differential equation (PDE) related problems.

3 Problem Definition

Our objective is to predict the target variable baseflow for each basin $m \in \{1, \dots, M\}$, and on each date $t \in \{1, \dots, T\}$, given input physical variables that drive the dynamics of the physical system. In detail, we use $\mathbf{X}_d = \{\mathbf{x}_{m,d}^t\}$ to represent dynamic climate input features for each location m on a specific date t , and use $\mathbf{X}_s$ to represent the hydrological conditions for each location. $\mathbf{X}_s$ include several hydrological features of all river basins such as soil features (more details provided in Section 5.1). We also regard $\mathbf{X} = \{\mathbf{x}_m^t\}$ as the complete input features, which contain both $\mathbf{X}_d$ and $\mathbf{X}_s$. Then, we aim to predict the corresponding target variables (i.e., baseflow) $\mathbf{Y} = \{y_m^t\}$.

Additionally, we leverage the simulation data generated by different physics-based equations (parameterizations) into the ML model training. The physics-based equations take the input features $\mathbf{X} = \{\mathbf{x}_m^t\}$ and simulate target variables based on known physical theory and a set of estimated parameters. Specifically, the physics-based equations simulate baseflow based on input features (e.g., rainfall and soil moisture), which can affect the discharge of groundwater directly. In this work, we selected K different sets of simulated data generated by using K different physics-based equations (that are commonly used by domain scientists). We represent each set of simulated target variables as $\tilde{\mathbf{Y}}_k = \{\tilde{y}_{m,k}^t\}$.

4 Method

In Fig. 1, we show the overall architecture of our proposed framework PGMTL. This framework is agnostic

of the specific predictive model. We adopt standard LSTM [9] as a base predictive model in this work but the proposed framework can be applied to other models (e.g., EA-LSTM [33]). In the proposed framework, we first use the data of all river basins to train a global model. Then we divide all the river basins into N different clusters according to their hydrological conditions. The global model is adjusted through a meta-learning process to find the optimal global initial model, which is then adapted to different clusters. In this process, we also introduce three different physics-guided meta-learning methods to incorporate the physical information into our model training and learn separate models for each cluster.

4.1 Physics-Guided Meta-Learning To help the predictive model better capture underlying physical relationships in baseflow dynamics, we introduce multiple sets of simulations generated by different physics-based equations to enhance the predictive model.

Hydrological scientists use different physics-based equations to simulate the baseflow dynamics as baseflow behaviors vary across river basins due to their distinct hydrological conditions [8]. These equations are created based on both known physical relationships and empirical analysis. Each physics-based equation often performs better on certain basins over other basins. This suggests that multiple sets of simulations created using different equations can have different degrees of contributions to modeling different river basins. To better utilize simulations to aid model prediction, we use the K-means clustering algorithm to divide all river basins into different clusters according to the hydrological conditions of river basins $\mathbf{X}_s$. Then, we consider each cluster of river basins to be a learning task, and we aim to build a model that can be quickly adapted to all the tasks. The river basins in each cluster are expected to share similar hydrological conditions, and thus can ben-

efit from the knowledge embedded in the same set of physical simulations.

Then we introduce new physics-guided meta-learning methods to integrate different simulations into the learning process for each task (i.e., a cluster of basins). Inspired by the MAML algorithm [28], we aim to identify a set of shared initial parameters Θ_0 such that the model can be slightly fine-tuned to fit each new task. Our method extends the MAML method in that we also introduce some new model components during the adaptation process to each task by leveraging auxiliary physical knowledge. Specifically, in addition to the set of shared initial parameters Θ_0 , we also create another set of distinct parameters Φ_i for each task i , which is used to capture the contributions from different sets of physical simulation data.

In particular, in the meta-learning process, we start from an initial global model with Θ_0 and fine-tune it to N different clusters separately using the training data $\{\mathbf{X}_i^{\text{tr}}, \mathbf{Y}_i^{\text{tr}}\}$ for each task. Such a fine-tuning process produces N sets of task-specific parameters Θ_i for $i = 1$ to N . In this process, we also train another set of distinct parameters Φ_i for N different clusters. These parameters Φ_i are used to create additional regularization or augmentation components to enhance the base model (parameterized by Θ_i) by leveraging physical knowledge (will be discussed in Section 4.2). This process can be represented by:

$$(4.1) \quad \Theta_i = \Theta_0 - \alpha \nabla_{\Theta} \mathcal{L}(f(\mathbf{X}_i^{\text{tr}}; \Theta, \Phi_i), \mathbf{Y}_i^{\text{tr}})|_{\Theta=\Theta_0},$$

where α is a learning rate of our proposed model, $f(\cdot)$ represents the mapping relation from input features to target variables defined by the PGMTL method, $\mathcal{L}$ represents the mean squared error (MSE) loss function, and Θ_0 is a set of shared initial parameters. Here we only show a one-step gradient descent process, but it can be generalized to multiple gradient steps. The one-step gradient is adopted in our test and achieves good performance.

Once gathering Θ_i for each cluster, we define the loss of meta-learning on an independent validation set of each task $\{\mathbf{X}_i^{\text{val}}, \mathbf{Y}_i^{\text{val}}\}$, optimizing Θ_0 and $\{\Phi_i\}_{i=1}^N$ as:

$$(4.2) \quad \min_{\Theta_0, \{\Phi_i\}} \mathcal{L}_{\text{meta}} = \sum_{i=1}^N \mathcal{L}(f(\mathbf{X}_i^{\text{val}}; \Theta_i, \Phi_i), \mathbf{Y}_i^{\text{val}}).$$

In our implementation, we first train a global predictive model using data from all tasks and obtain its parameters Θ_g . Note that this global model does not use parameters Φ . This is because different sets of basins may be better simulated by different physics-based equations, and thus it is difficult to use a single set of parameters Φ to capture the contribution of different physics-based equations for all the basins. Then

we use Θ_g to initialize Θ_0 and run the meta-learning process. After obtaining the optimal initial parameters Θ_0 and the other parameters Φ_i for each task i , we can test the model by fine-tuning the model for each cluster and update Θ_0 to be Θ_i . The parameters Φ_i are fixed during the fine-tuning process in the testing phase.

4.2 Methods of Integrating Simulations We consider three different methods for incorporating physical simulations (created using multiple physical equations) into the PGMTL framework: (i) augmentation, (ii) attention mechanism, and (iii) contrastive learning. These methods are described in the following:

4.2.1 Augmentation The meta-learning loss in Eq. 4.2 requires the predicted target variables and true observations. Here we represent the predicted target variables as $\hat{\mathbf{Y}} = \{\hat{y}_m^t\}$, and the loss function of standard LSTM is defined using true observations $\mathbf{Y} = \{y_m^t\}$ that are available at certain time steps for certain locations, described as:

$$(4.3) \quad \mathcal{L}_{\text{LSTM}}(\hat{\mathbf{Y}}, \mathbf{Y}) = \frac{1}{|\mathbf{Y}|} \sum_{\{(m,t)|y_m^t \in \mathbf{Y}\}} (y_m^t - \hat{y}_m^t)^2.$$

Next, in order to incorporate the simulations generated from different physics-based equations into our proposed model, we also calculate the loss between each set of simulated target variables $\tilde{\mathbf{Y}}_k = \{\tilde{y}_{m,k}^t\}$ and predicted target variables $\hat{y}_m^t$, which is shown as:

$$(4.4) \quad \mathcal{L}_{\text{phy}}(\hat{\mathbf{Y}}, \tilde{\mathbf{Y}}_k) = \frac{1}{|\tilde{\mathbf{Y}}_k|} \sum_{\{(m,t)|\tilde{y}_{m,k}^t \in \tilde{\mathbf{Y}}_k\}} (\tilde{y}_{m,k}^t - \hat{y}_m^t)^2.$$

To train the predictive model while preserving the consistency with simulated data, we combine these two loss functions together and obtain the complete loss function, as follows:

$$(4.5) \quad \mathcal{L}_{\text{AUG}} = \mathcal{L}_{\text{LSTM}}(\hat{\mathbf{Y}}, \mathbf{Y}) + \sum_k \lambda_k \mathcal{L}_{\text{phy}}(\hat{\mathbf{Y}}, \tilde{\mathbf{Y}}_k),$$

where λ_k represents the weight of k^{th} set of simulation data. In particular, this new loss $\mathcal{L}_{\text{AUG}}$ is used to replace the loss in Eq. 4.1 while we still use the standard supervised loss in Eq. 4.2. Here $\Phi = \{\lambda_k\}$ and is essentially the parameters of the loss function instead of the predictive model f . These parameters $\{\lambda_k\}$ need to be tuned separately for each task. It is also worthwhile to mention that the physical loss $\mathcal{L}_{\text{phy}}$ can be evaluated for a larger set of data samples for which observations are not available.

4.2.2 Attention Mechanism We also adapt the attention mechanism [34] to combine the predicted

target variables $\hat{y}_m^t$ with the simulated target variables $\tilde{\mathbf{Y}}_k = \{\tilde{y}_{m,k}^t\}$ to obtain the final predicted target variables $\mathbf{Y}_f = \{\hat{y}_{m,f}^t\}$, which can be represented by the following equation:

$$(4.6) \quad \hat{y}_{m,f}^t = w_0^t \hat{y}_m^t + w_k^t \sum_{k=1}^K \tilde{y}_{m,k}^t,$$

where w_k^t ($k = 0, 1, \dots, K$) are weights for $\hat{y}_m^t$ and $\tilde{y}_{m,k}^t$. These sets of weights w_k^t come from another LSTM model. In particular, we use a separate LSTM model to produce hidden representation $\{\mathbf{h}_a^t\} = g(\mathbf{x}^t)$ on each date t . Our goal is to use the attention mechanism by a fully connected layer to convert $\{\mathbf{h}_a^t\}$ to a set of weights w_k^t on each date t corresponding to the predicted target variables $\hat{y}_m^t$ and the simulated target variables $\tilde{y}_{m,k}^t$. Once we obtain a set of weights for $\hat{y}_m^t$ and $\tilde{y}_{m,k}^t$, we normalize them into attention weights w_k^t via softmax function. More formally, the attention weights can be represented by:

$$(4.7) \quad w_k^t = \frac{\mathbf{W}_k \mathbf{h}_a^t + \mathbf{b}_k}{\sum_{k'} \mathbf{W}_{k'} \mathbf{h}_a^t + \mathbf{b}_{k'}},$$

where $\mathbf{W}_k$ and $\mathbf{b}_k$ are attention model parameters. These parameters and the parameters introduced in the additional LSTM $g(\cdot)$ form the parameter set Φ .

Using the predictions produced by Eq. 4.6, we can measure the complete loss as:

$$(4.8) \quad \mathcal{L}_{\text{ATN}}(Y_f, \mathbf{Y}) = \frac{1}{|\mathbf{Y}|} \sum_{\{(m,t)|y_m^t \in \mathbf{Y}\}} (y_m^t - \hat{y}_{m,f}^t)^2,$$

This loss $\mathcal{L}_{\text{ATN}}$ is used in both the inner meta-update process (Eq. 4.1) and the outer update process (Eq. 4.2).

4.2.3 Contrastive Learning As the augmentation method did, we calculate the loss $\mathcal{L}_{\text{LSTM}}(\hat{\mathbf{Y}}, \mathbf{Y})$ between the predicted target variables and true observations. Then, we will also further leverage the knowledge extracted from simulation data by exploring the relationships between the observation data and different sets of simulation data. Here we will introduce a new loss function to capture this relationship.

After gathering the predicted target variables $\hat{y}_m^t$ and simulated target variables $\tilde{y}_{m,k}^t$, we define a similarity mapping $\hat{y}_m^t \rightarrow \tilde{y}_{m,k}^t$ for each $k=1$ to K using the distance between $\hat{y}_m^t$ and $\tilde{y}_{m,k}^t$. Once we obtain the similarity values for all sets of simulation data, we normalize the obtained similarity values and convert them into a distribution $\mathcal{Q}(\hat{y}_m^t \rightarrow \tilde{y}_{m,k}^t)$ via a softmax function. More formally, this can be expressed as:

$$(4.9) \quad \mathcal{Q}(\hat{y}_m^t \rightarrow \tilde{y}_{m,k}^t) = \frac{\exp(-|\tilde{y}_{m,k}^t - \hat{y}_m^t|)}{\sum_{k'} \exp(-|\tilde{y}_{m,k'}^t - \hat{y}_m^t|)}.$$

We aim to ensure that the patterns extracted from observation data are similar to the sets of simulation data created by physical equations suitable for modeling the target basin system while staying different from other sets of simulations. Specifically, we define a contrastive loss based on the entropy of the similarity probability, as follows:

$$(4.10) \quad \mathcal{L}_{\text{ctr}} = -\frac{1}{|\mathbf{Y}|} \sum_{\{y_m^t \in \mathbf{Y}\}} \sum_{k=1}^K \mathcal{Q}(\hat{y}_m^t \rightarrow \tilde{y}_{m,k}^t) \log \mathcal{Q}(\hat{y}_m^t \rightarrow \tilde{y}_{m,k}^t).$$

Combining the contrastive loss and the loss $\mathcal{L}_{\text{LSTM}}(\hat{\mathbf{Y}}, \mathbf{Y})$, we get the complete loss, as follows:

$$(4.11) \quad \mathcal{L}_{\text{CTR-all}} = \mathcal{L}_{\text{LSTM}}(\hat{\mathbf{Y}}, \mathbf{Y}) + \gamma \mathcal{L}_{\text{ctr}},$$

where γ is a hyper-parameter.

The contrastive learning method adapts a new regularizer based on different sets of simulations but unlike the previous two methods, without involving any additional model parameters Φ . The meta-learning process is conducted by updating Θ_0 to Θ_i with a new loss function $\mathcal{L}_{\text{CTR-all}}$. This loss $\mathcal{L}_{\text{CTR-all}}$ is used in the inner meta update process (Eq. 4.1) while the validation loss in Eq. 4.2 remains the standard supervised loss. This is the same with the augmentation method.

4.3 Implementation Details We clarify the implementation details of our framework in the following:

We first utilize input feature $\mathbf{X}$, including dynamic climate input features $\mathbf{X}_d$ and hydrological condition $\mathbf{X}_s$, to train a global model, and obtain a set of parameters of Θ_g . Second, we adapt Θ_g to initialize shared parameters Θ_0 , and randomly initialize another set of parameters Φ_i . We continue to use the physics-guided meta-learning method to train each task in a random order multiple times. Then, we obtain a set of shared parameters Θ_0 and another set of distinct parameters Φ_i for each task. Lastly, we adapt the learned model to each cluster by fine-tuning Θ_0 into Θ_i while fixing Φ_i , and use the updated parameters $\{\Theta_i, \Phi_i\}$ to predict target variables.

Second, we obtain simulations from physics-based equations. We select two physics-based equations which are widely used in the hydrology domain to simulate baseflow [8], as:

$$(4.12) \quad \begin{aligned} \tilde{\mathbf{Y}}_1 &= b(s^p), \\ \tilde{\mathbf{Y}}_2 &= c * \exp(b(s - s_{\max})), \end{aligned}$$

where c is hyper-parameter, b and p are equation parameters, s is the soil moisture, and $s_{\max}$ is the maximum value of soil moisture in a certain river basin. The proposed method can be applied to other baseflow equations as summarized in [8].

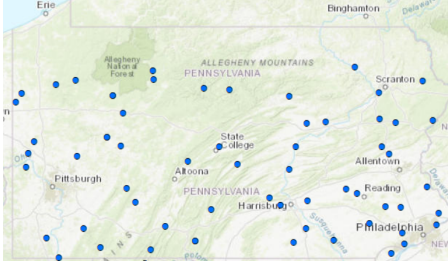


Figure 2: shows the parts of the selected river basin’s distribution in PA, the blue circles represent the locations of river basins.

To find the appropriate pair value of parameters b and p , we first use the gradient descent method to fit both equations and find the suitable initial value of parameters b and p for each river basin. Then we take them as the initial values and adjust them 300 times using a grid search, each adjustment with an amplitude of 0.05. Finally, we can find the best pair value of parameters b and p , which can bring simulated target variables generated by these two equations closest to the true observation. Such a calibration process is conducted using only the training samples.

5 Experiments

5.1 Dataset *Climate Driver and Observation:* We study 60 river basins in Pennsylvania, and the distribution of a subset of these river basins can be seen in Fig. 2. We use the dynamic input feature on a daily scale from January 01, 1987, to July 27, 2016 (a total of 10,800 days). The input dynamic features are taken from the NCA-LDAS dataset [35], including 20 variables on energy fluxes (e.g., radiation), snow/rain precipitation rate, evapotranspiration, potential evaporation rate, snowmelt, soil moisture in different depths, some features of wind speed, humidity and pressure. We also have static hydrological conditions of river basins from the GAGES-II dataset [36], which includes 40 dimensions such as soil properties (e.g., bulk density), stream geometry (e.g., stream lengths), and dam-related features (e.g., dam density). The baseflow time series obtained from the one-parameter digital filter method [15] of the HS approach is used as the labels in our experiment. This method has been shown in prior work to provide high-fidelity baseflow estimates [21].

Simulation data: For the simulation data, we generate two sets of different simulation data by two different physics-based equations (Eq. (4.12)). We use these two physics-based equations to generate the simulated data over the same study period of 10,800 days.

5.2 Experiment Settings Firstly, we compare the performance of our proposed PGMTL method against

Table 1: Predictive NSE [37] performance for baseflow prediction, with columns 2-4 showing the results using 0.5%, 1%, and 100% training data, respectively.

Method	0.5%	1%	100%
LSTM	0.645	0.821	0.875
LSTM-M	0.817	0.836	0.880
LSTM-C	0.839	0.841	0.883
EA-LSTM	0.778	0.821	0.871
EA-LSTM-M	0.816	0.865	0.867
EA-LSTM-C	0.836	0.870	0.891
PGMTL-Augmentation	0.857	0.875	0.901
PGMTL-Attention	0.864	0.877	0.908
PGMTL-Contrastive	0.859	0.876	0.921

several existing methods that have been widely used in the hydrology domain. Specifically, we implement our three proposed methods: PGMTL-Augmentation, PGMTL-Attention, PGMTL-Contrastive¹, and also implement LSTM [9], EA-LSTM [33] as baselines. In particular, EA-LSTM is a specially designed variant of LSTM, which uses hydrological conditions to modify the input gate of the LSTM cell. This is to simulate how hydrological conditions can be used to filter the impact from the current climate input (e.g., rainfall and radiation) to the state of the hydrological system. In addition, to better verify the effectiveness of our proposed clustering and meta-learning methods, we further implement four extra baselines LSTM-M, LSTM-C, EA-LSTM-M, and EA-LSTM-C, described below:

Baseline methods with MAML [28] (LSTM-M and EA-LSTM-M): We compare to the standard MAML method by implementing the MAML method using the base model of LSTM and EA-LSTM.

Baseline methods with MAML + Clustering (LSTM-C and EA-LSTM-C): We also implement MAML at the clustering level (using the proposed clustering mechanism) and combine it with both LSTM-M and EA-LSTM-M. The comparison between our proposed PGMTL model and these two baselines (LSTM-C and EA-LSTM-C) shows the effect of incorporating physical relationships in the model adaptation process.

In the following experiments, we use data from January 01, 1987, to September 18, 2006, for training, and then measure the testing performance on data from September 18, 2006, to July 27, 2016. Specifically, during the processing of our proposed meta-learning methods, we use data from January 01, 1987, to November 19, 1996 as training (totally 3,600 days), and data from November 19, 1996, to September 18, 2006, as validating (totally 3,600 days).

¹The source code for the PGMTL-based models and dataset presented in this study are available online at the link: <https://drive.google.com/drive/folders/1j-80wmk71a7XKy11B0DKDuqyM5L4Gvvr?usp=sharing>

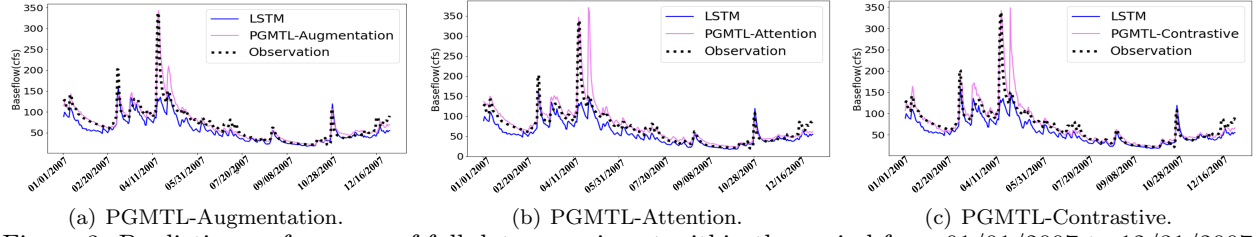


Figure 3: Predictive performance of full data experiment within the period from 01/01/2007 to 12/31/2007.

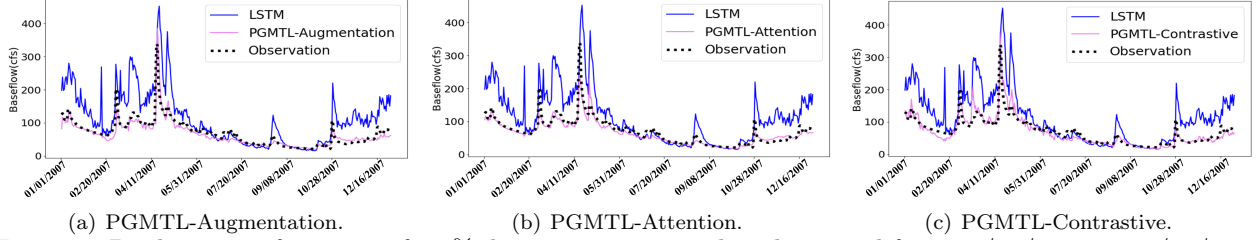


Figure 4: Predictive performance of 0.5% data experiment within the period from 01/01/2007 to 12/31/2007.

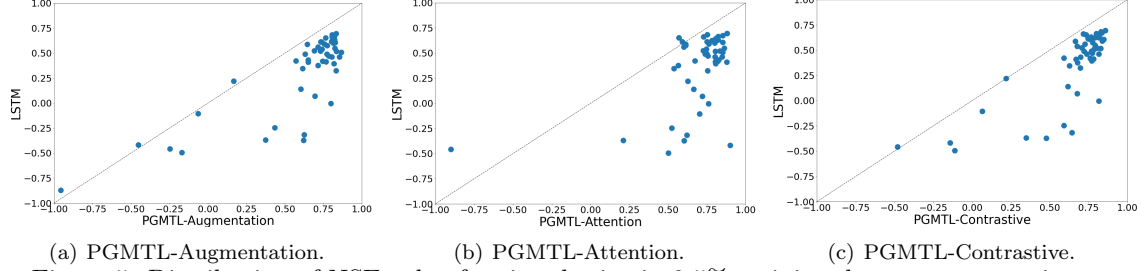


Figure 5: Distribution of NSE value for river basins in 0.5% training data sparse experiment.

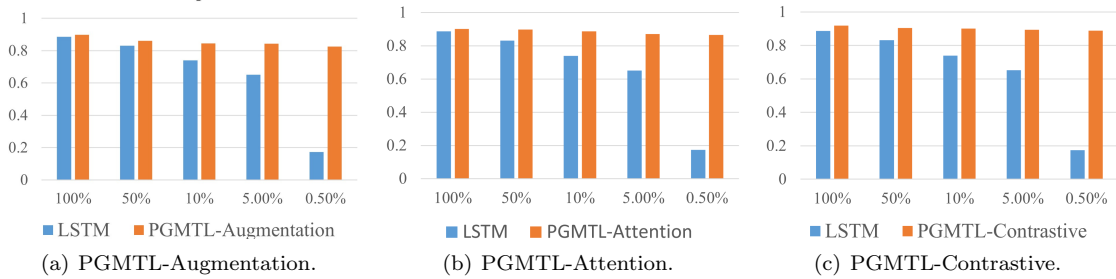


Figure 6: Transferring Experiment's NSE Comparisons between PGMTL methods and baseline LSTM model.

5.3 Prediction Performance In the following, we compare our proposed model with baselines by using the Nash-Sutcliffe model efficiency coefficient (NSE) [37]. NSE is widely used to assess the predictive performance of hydrological models. The value of NSE ranges in $(-\infty, 1]$, and the higher value indicates better performance.

5.3.1 Full Data Experiment We report the performance of each method using data from 60 basins in Table 1 (the last column). We have several observations: (1) By comparing LSTM and LSTM-M, we show the improvement by using the MAML method. Also, by comparing LSTM-M and LSTM-C, we show the effectiveness of using the clustering mechanism in our proposed methods. We can also obtain a similar conclusion by comparing the EA-LSTM-based baseline

methods. (2) When comparing our proposed PGMTL methods with baseline methods, our methods perform better and produce the highest NSE. It means that incorporating the simulation data into the ML model can bring significant improvement in terms of NSE. (3) We can observe similar performance between LSTM-based and EA-LSTM-based methods in this experiment. It means LSTM is sufficient to directly learn how to incorporate hydrological conditions into the prediction. Hence, we use LSTM as the base model for implementing PGMTL-based methods.

On the other hand, Figs. 3 (a)-(c) show that the LSTM model often underestimates the baseflow of river basins, especially in the Spring period. Our proposed PGMTL model, regardless of which adaptation methods are used in meta-learning, can obtain a better pre-

diction performance, especially in the Spring when the discharge of groundwater is larger.

5.3.2 Sparse Data Experiment To simulate the difficulty of data collection in the real world, we also conduct a group of sparse tests. In these tests, we chose to use only 1% and 0.5% of the training data to train our proposed model and baseline models, to prove that our proposed PGMTL model still has a great advantage when the training data is limited.

In this sparse experiment, we have new observations from Table 1: (1) As training data becomes sparser, the improvement of our proposed methods to the model becomes more significant. Specifically, considering the experiment using only the 0.5% training data, we can observe a huge improvement from the global pre-training model (LSTM) to our proposed models. It also shows that our methods are still highly adaptable in the case of sparse observation data. (2) Similar to the previous full data experiment, we show the effectiveness of proposed methods against LSTM, LSTM-M, and LSTM-C. (3) By comparing the LSTM and EA-LSTM methods, we can observe a significant difference in using only 0.5% training data, which can suggest that the EA-LSTM model would be more suitable than the LSTM model if data is extremely sparse. In future extensions, we may use the EA-LSTM model as a benchmark instead of LSTM in this extreme condition.

At the same time, from Figs. 4 (a)-(c), we can see that the LSTM model produces much worse predictions, especially for time periods with baseflow oscillations, if it uses extremely limited training data. As for the PGMTL-based methods, although their performance is worse than the performance obtained using sufficient training data, they still have a great advantage compared with the LSTM model. On the other hand, Figs. 5 (a)-(c) show the comparison between the PGMTL-based methods and the LSTM model for each basin in terms of the NSE value. These results suggest that more than 90% river basins will get better performance if using our proposed methods, which confirms that our proposed model has significant superiority over the baseline model.

5.3.3 Transferring Experiment In addition, we also design a transferring experiment to test the PGMTL-based methods in real-world scenarios in which the available training data are highly localized in certain basins while other basins only have limited training data. We plan to test whether the proposed model and the standard ML model can use the information of other river basins with sufficient data to improve the prediction of other river basins with limited training data.

In this experiment, we randomly select half of the river basins in each cluster and randomly sparsify their training data by 50%, 10%, 5%, and 0.5%, respectively, and then measure the testing performance on these selected basins. The training data for other basins still remain the same. We show the NSE performance between our proposed methods and baseline model LSTM in Figs. 6 (a)-(c). From these figures, we can conclude that: (1) When the training data is sufficient (e.g. over 50% training data), both the proposed PGMTL methods and the baseline LSTM model can achieve a good NSE score, and the performance decrease only slightly as we reduce the training data. It can be seen that the decreasing degree of NSE value of our methods is still slightly slower than the LSTM model. (2) When reducing the amount of training data to less than 10%, it is clear that LSTM performance starts to decrease much faster while the decreasing degree of our methods is much slower than the baseline LSTM model. (3) In the extreme case with only 0.5% training data, the LSTM model has bad performance. This indicates that the LSTM model cannot fully leverage the information from other well-observed basins. In contrast, although our proposed PGMTL-based methods have worse performance with sparser data, it performs much better than LSTM, which confirms that our methods can effectively transfer knowledge from well-observed basins to poorly-observed basins. All of the discoveries also prove the superiority of the proposed methods, which can be easily adapted to real-world conditions.

6 Conclusion

In this paper, we develop a new data-driven method PGMTL that integrates scientific knowledge from simulation data into the model adaptation process. We also introduce a clustering mechanism based on the hydrological conditions of basins. Under the obtained clustering structure, our method can capture the contribution level of multiple physics-based equations (i.e., physical equations) for different clusters of the river basins under different hydrological environments. At the same time, we come up with three different machine learning methods to transfer physical knowledge from simulation data to the model adaptation process in meta-learning. Our experiments in real river basin data have demonstrated the effectiveness of these methods in improving baseflow prediction even with sparse or localized data. Moreover, we have shown in a synthetic data experiment that our proposed PGMTL can reveal the contribution of each physics-based equation under different scenarios. In future work, we will continue to explore a larger variety of physics-based equations, furthermore, evaluate their contributions in larger regions. We also will

propose new extensions to dynamically learn the clustering structure from hydrological conditions while also optimizing the meta-learning performance.

Acknowledgements

This work was supported by NSF awards 2147195, 2105133, and 2126474, NASA award 80NSSC22K1164, the USGS awards G21AC10207, G21AC10564, and G22AC00266, Google's AI for Social Good Impact Scholars program, the DRI award at the University of Maryland, and CRC at the University of Pittsburgh.

References

- [1] Wilfried Brutsaert. Long-term groundwater storage trends estimated from streamflow records: Climatic perspective. *WRR*, 2008.
- [2] Paul A. Dirmeyer. Using a global soil wetness dataset to improve seasonal climate simulation. *Journal of Climate*, 2000.
- [3] Derly Gómez et al. Empirical rainfall-based model for defining baseflow and dynamical water use rights. *River Research and Applications*, 2020.
- [4] Christine A Rumsey et al. Regional Studies Regional scale estimates of baseflow and factors influencing baseflow in the Upper Colorado River Basin. 2015.
- [5] Peter R. Furey et al. Tests of two physically based filters for base flow separation. *WRR*, 2003.
- [6] Katie Price. Effects of watershed topography, soils, land use, and climate on baseflow hydrology in humid regions: A review. *PPG*, 2011.
- [7] Christopher J. Duffy. A two-state integral-balance model for soil moisture and groundwater dynamics in complex terrain. *WRR*, 1996.
- [8] L. M. Tallaksen. A review of baseflow recession analysis. *Journal of Hydrology*, 1995.
- [9] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 1997.
- [10] Frederik Kratzert et al. Toward improved predictions in ungauged basins: Exploiting the power of machine learning. *WRR*, 2019.
- [11] Xiaowei Jia et al. Physics guided rnns for modeling dynamical systems: A case study in simulating lake temperature profiles. In *SDM*, 2019.
- [12] Zach Moshe et al. Hydronets: Leveraging river structure for hydrologic modeling. 2020.
- [13] Shengyu Chen et al. Heterogeneous stream-reservoir graph networks with data assimilation. In *ICDM*, 2021.
- [14] Rory M. Cowie et al. Sources of streamflow along a headwater catchment elevational gradient. *Science*, 1990.
- [15] V Lyne et al. Stochastic time-variable rainfall-runoff modelling. In *Institute of Engineers Australia National Conference*, 1979.
- [16] Kyoung Jae Lim et al. Automated web gis based hydrograph analysis tool, what 1. *JAWRA*, 2005.
- [17] Klaus Eckhardt. How to construct recursive digital filters for baseflow separation. *Hydrological Processes: An International Journal*, 2005.
- [18] Ven Te Chow. *Applied hydrology*. Tata McGraw-Hill Education, 2010.
- [19] Ronald A Sloto et al. Hysep: A computer program for streamflow hydrograph separation and analysis. *Water-resources investigations report*, 1996.
- [20] Institute of Hydrology. Low flow studies report no. 1: Research report. 1980.
- [21] Sebastian J Gnnann et al. Is there a baseflow budyko curve? *WRR*, 2019.
- [22] Anuj Karpatne et al. Physics-guided neural networks (pgnn): An application in lake temperature modeling. *arXiv*, 2017.
- [23] Jordan S Read et al. Process-guided deep learning predictions of lake water temperature. *WRR*, 2019.
- [24] Dehao Liu et al. Multi-fidelity physics-constrained neural network and its application in materials modeling. *Journal of Mechanical Design*, 2019.
- [25] Jared Willard et al. Integrating physics-based modeling with machine learning: A survey. *arXiv*, 2020.
- [26] Paul C Hanson et al. Predicting lake surface water phosphorus dynamics using process-guided machine learning. *Ecological Modelling*, 2020.
- [27] Huaxiu Yao et al. Hierarchically structured meta-learning. PMLR, 2019.
- [28] Chelsea Finn et al. Model-agnostic meta-learning for fast adaptation of deep networks. *arXiv*, 2017.
- [29] Xiaowei Jia et al. Physics-guided machine learning from simulation data: An application in modeling lake and river systems. In *ICDM*, 2021.
- [30] Xu Liu et al. A novel meta-learning initialization method for physics-informed neural networks. *arXiv*, 2021.
- [31] Shengyu Chen, Jacob A Zwart, and Xiaowei Jia. Physics-guided graph meta learning for predicting water temperature and streamflow in stream networks. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2752–2761, 2022.
- [32] Maziar Raissi et al. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 2019.
- [33] Frederik Kratzert et al. Benchmarking a catchment-aware long short-term memory network (lstm) for large-scale hydrological modeling. *Hydrology and Earth System Sciences Discussions*, 2019.
- [34] Zhaoyang Niu et al. A review on the attention mechanism of deep learning. *Neurocomputing*, 2021.
- [35] Michael F. Jasinski et al. Nca-ldas: Overview and analysis of hydrologic trends for the national climate assessment. *Journal of Hydrometeorology*, 2019.
- [36] James A. Falcone. Nages-ii: Geospatial attributes of gages for evaluating streamflow. 2011.
- [37] Wikipedia contributors. Nash-sutcliffe model efficiency coefficient, 2021.