Assessment of Several Regression Based Machine Learning Approaches in Runoff Prediction

Sadegh Sadeghi Tabas
Department of Civil Engineering
CUID: C93338799
sadeghs@clemson.edu

CPSC8420 Project Milestone

1 Introduction

1.1 Problem Definition

Watershed simulations help understand the past and current state of rainfall-runoff processes in the basin and provide a way to explore the implications of management and planning decisions and imposed changes (such as land use change, climate change). In complex environmental systems such as the coastal plain watersheds, very significant modeling efforts have gone through lumped and physical based simulations (Sadeghi-Tabas et al., 2017; Samadi et al., 2020, 2017; Samadi and Meadows, 2017). Indeed, in pursuit of improved accuracy, there is a plethora of hydrological tools that have been enhanced and implemented for complex rainfall-runoff simulations. In the realm of process-based hydrologic modeling, time-series machine learning (ML) or data-driven algorithms have been recently utilized to improve short- and long-term streamflow predictions at various scales and domains (e.g., Fang et al., 2017; Kratzert et al., 2018; Shen, 2018; Shen et al., 2018; Wu et al., 2018).

Data-driven algorithms attempt to estimate the mapping function (f) from the input variables (x) to numerical or continuous output variables (y) and to understand time-series natural temporal ordering across long-term records. These methods can directly learn patterns hidden in time-series data, without requiring manually-designed features or making strong physical assumptions (Kasiviswanathan et al., 2016; LeCun et al., 2015; Schmidhuber, 2015) and are good at handling large datasets with high dimensionality and heterogeneous feature types. Many studies have demonstrated that ML models can outperform other state-of-the-art techniques in hydrologic simulation (Yang et al., 2020). Among various ML algorithms, regression-based models such as Support Vector Machines (SVM), Random Forest (RF), Multi-layer Perceptron (MLP), and Decision Tree models have widely applied for streamflow simulations and forecasting.

1.2 Literature Review

1.3 Motivation and Novelty

In this study we are going to predict daily runoff for Stevens Creek Basin, SC, USA, using various state-of-the-art ML methods as data-driven approaches. Data-driven approach can be appropriate tools for runoff simulation due to the complexity of modeling rainfall-runoff processes, which makes using a physical model difficult. This study investigated and compared 5 different ML algorithms including Linear Regression, Lasso Regression, Ridge Regression, Multilayer Perceptron (MLP), and Support Vector Machine (SVM) in runoff prediction. To do so, first the preprocessing of the available dataset has been done in order to impute the missing values. Then the mentioned data drivon methods trained and tested using various performance assessment criteria. The detailed description of process and the challenges in this research project is presented in the following sections including metodology and results.

2 Methodology

2.1 Case study: Stevens Creek Basin

I have selected the Stevens Creek Basin located in South Carolina, US (USGS Guage ID: 02196000). The underlying data for case study is retrieved from the CAMELS data set which are provided by Newman et. al (2015) and are publicly available at the *UCAR website*. The data set contains catchment meteorological forcing data and observed discharge at the daily timescale. The dataset is consist of precipitation, shortwave downward radiation, maximum and minimum temperature, and vapor pressure. In this research all meteorological forcing data as well as antecedent day runoff considered as input to each ML method.

2.2 Simulation Models

Machine learning algorithms are powerful models that first consider the learning styles that an algorithm can adopt. These approaches can be divided into 3 broad categories(i) supervised learning, (ii) unsupervised learning, and (iii) reinforcement learning. Supervised learning is useful in cases where a property (label) is available for a certain dataset (training set) but is missing and needs to be predicted for other instances. Unsupervised learning is useful in cases where the challenge is to discover implicit relationships in a given unlabeled dataset (items are not pre-assigned). Reinforcement learning falls between these 2 extremes—there is some form of feedback available for each predictive step or action, but no precise label or error message. In this chapter, we focused on using several supervised learning models as regression-based approaches to simulate daily runoff values. As I mentioned in the project proposal I am going to implement supervised ML methods to simulate runoff at time step t using mentioned meteorological forcing data (at time step t) as well as antecedent day runoff.

2.2.1 Linear Regression

2.2.2 Ridge Regression

2.2.3 Lasso Regression

2.2.4 Multilayer Perceptron (MLP)

2.2.5 Support Vector Machine (SVM)

2.3 Performance Assessment Criteria

Because no one evaluation metric can fully capture the consistency, reliability, accuracy, and precision of a streamflow model, it was necessary to use a variety of performance metrics for model benchmarking (Hoshin Vijai Gupta et al., 1998). The metrics for model evaluation are the Nash–Sutcliffe efficiency (NSE; Nash and Sutcliffe, 1970), the three decompositions following Hoshin V Gupta et al. (2009) which are the correlation coefficient of the observed and simulated discharge (r), the variance bias (α) and the total volume bias (β). These three metrics are combined and presented in the Kling-Gupta efficiency (KGE; Gupta et al., 2009) metric presented in equation 2. And the third critrion to evaluate the erros in simulations is root mean squared error (RMSE) presented in equation 3. in these equation Q_s is the simulated runoff, Q_o observed runoff.

$$NSE = 1 - \frac{\sum_{t=1}^{t}(Q_s^t - Q_o^t)^2}{\sum_{t=1}^{t}(Q_o^t - \bar{Q}_o)^2} \tag{1}$$

$$KGE = 1 - \sqrt{(cc-1)^2 + (\alpha-1)^2 + (\beta-1)^2} \tag{2}$$

$$RMSE = \sqrt{\sum_{t=1}^{n} \frac{(Q_s^t - Q_o^t)^2}{n}} \tag{3}$$

3 Preliminary Results

As of today, a widespread calibration strategy for the ML models is to subdivide the data into two parts, referred to as training and test datasets. The first split is used to derive the parametrization of the models (calibration in the context of hydrologic simulation) and the remainder of the data to diagnose the actual performance (validation in the context of hydrologic simulation). We used 20 years of the available data from Jan 01, 1984 to Dec 31, 2003 as training data and the next 6 years as the independent test period (Jan 01, 2003 to Dec 31, 2009). The training period is so important to get a good actual result because a small division of dataset as training period would lead to a high risk of overfitting. Figure 1 shows the observed and simulated discharge of various ML methods during test period.

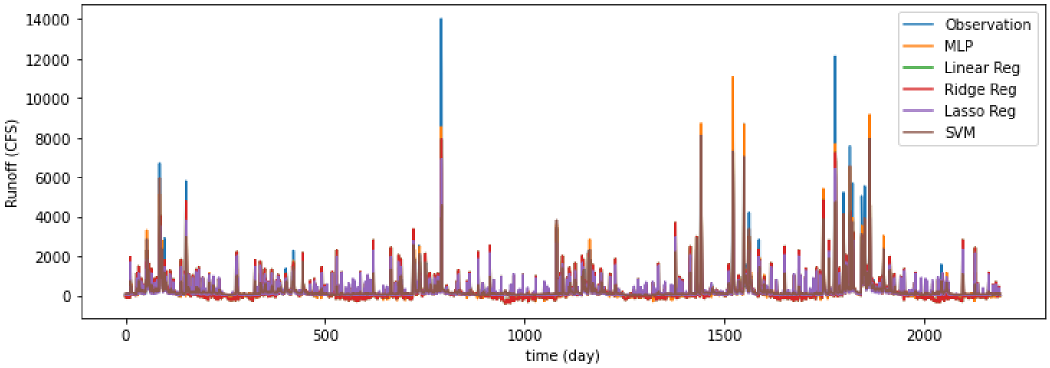

Figure 1: Runoff simulations using different ML methods as well as observation values for the test period (Jan 01, 2004 - Dec 31, 2009)

The visualization of streamflow prediction shows the MLP and SVM models performed significantly better than the others (Table 1). Various metrics are performed and presented in Table 1 for performance assessment.

Table 1: The calculated performance criteria for the simulated runoff in the test period

Models	NSE	RMSE	KGE
Linnear Regression	0.65	692.41	0.68
Lasso Regression	0.62	722.45	0.58
Ridge Regression	0.65	692.41	0.68
MLP	**0.8**	**530.96**	**0.84**
SVM	0.72	616.39	0.67

It is well-known that machine learning models could fail in simulating streamflows from only meteorological variables in the absence of antecedent streamflow values. The main reason for this could be low and lagged relationships between streamflow and meteorological variables (Tongal and Booij, 2018). To overcome this inefficiency I am going to first simulate the runoff (t-1) using a feed forward neural network algorithm (I will use just runoff observations to train FFNN and simulate antecedent day runoff), and then use different ML algorithms to simulate runoff (t).

4 Next Steps

In the final report I am going to:

- Complete the methodology part for different algorithms.
- As I mentioned that it is not possible to have successful simulation of runoff (t) using ML methods without the antecedent day runoff (t-1), so for the final report I am going to First simulate the runoff (t-1) using a feed forward neural network algorithm (I am going to train FFNN using just runoff observations), and then use different ML algorithms to simulate runoff (t).

- Complete the results section.
- Revise different sections of the report and update the reference section.
- Talk about trial and error process and model's settings to select the best parameters for some models
- I'll add a comprehensive explaination for each section in the final report (as I have to limit it to 3 pages in the milestone).

References

[1] K. Fang, C. Shen, D. Kifer, and X. Yang. Prolongation of smap to spatiotemporally seamless coverage of continental us using a deep learning neural network. *Geophysical Research Letters*, 44(21):11–030, 2017.

[2] H. V. Gupta, H. Kling, K. K. Yilmaz, and G. F. Martinez. Decomposition of the mean squared error and nse performance criteria: Implications for improving hydrological modelling. *Journal of hydrology*, 377(1-2):80–91, 2009.

[3] H. V. Gupta, S. Sorooshian, and P. O. Yapo. Toward improved calibration of hydrologic models: Multiple and noncommensurable measures of information. *Water Resources Research*, 34(4):751–763, 1998.

[4] K. Kasiviswanathan, J. He, K. Sudheer, and J.-H. Tay. Potential application of wavelet neural network ensemble to forecast streamflow for flood management. *Journal of Hydrology*, 536:161–173, 2016.

[5] F. Kratzert, D. Klotz, C. Brenner, K. Schulz, and M. Herrnegger. Rainfall–runoff modelling using long short-term memory (lstm) networks. *Hydrology and Earth System Sciences*, 22(11):6005–6022, 2018.

[6] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[7] Z. Liu, W. Xu, J. Feng, S. Palaiahnakote, T. Lu, et al. Context-aware attention lstm network for flood prediction. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 1301–1306. IEEE, 2018.

[8] J. E. Nash and J. V. Sutcliffe. River flow forecasting through conceptual models part i—a discussion of principles. *Journal of hydrology*, 10(3):282–290, 1970.

[9] A. Newman, M. Clark, K. Sampson, A. Wood, L. Hay, A. Bock, R. Viger, D. Blodgett, L. Brekke, J. Arnold, et al. Development of a large-sample watershed-scale hydrometeorological data set for the contiguous usa: data set characteristics and assessment of regional variability in hydrologic model performance. *Hydrology and Earth System Sciences*, 19(1):209, 2015.

[10] S. Sadeghi-Tabas, S. Samadi, B. Zahabiyoun, et al. Application of bayesian algorithm in continuous streamflow modeling of a mountain watershed. *European Water*, 57:101–108, 2017.

[11] S. Samadi and M. Meadows. The transferability of terrestrial water balance components under uncertainty and nonstationarity: A case study of the coastal plain watershed in the southeastern usa. *River Research and Applications*, 33(5):796–808, 2017.

[12] S. Samadi, M. Pourreza-Bilondi, C. Wilson, and D. Hitchcock. Bayesian model averaging with fixed and flexible priors: Theory, concepts, and calibration experiments for rainfall-runoff modeling. *Journal of Advances in Modeling Earth Systems*, 12(7):e2019MS001924, 2020.

[13] S. Samadi, D. Tufford, and G. Carbone. Assessing parameter uncertainty of a semi-distributed hydrology model for a shallow aquifer dominated environmental system. *JAWRA Journal of the American Water Resources Association*, 53(6):1368–1389, 2017.

[14] J. Schmidhuber. Deep learning in neural networks: An overview. *Neural networks*, 61:85–117, 2015.

[15] C. Shen. A transdisciplinary review of deep learning research and its relevance for water resources scientists. *Water Resources Research*, 54(11):8558–8593, 2018.

[16] C. Shen, E. Laloy, A. Elshorbagy, A. Albert, J. Bales, F.-J. Chang, S. Ganguly, K.-L. Hsu, D. Kifer, Z. Fang, et al. Hess opinions: Incubating deep-learning-powered hydrologic science advances as a community. *Hydrology and Earth System Sciences (Online)*, 22(11), 2018.

[17] H. Tongal and M. J. Booij. Simulation and forecasting of streamflows using machine learning models coupled with base flow separation. *Journal of hydrology*, 564:266–282, 2018.

[18] S. Yang, D. Yang, J. Chen, J. Santisirisomboon, W. Lu, and B. Zhao. A physical process and machine learning combined hydrological model for daily streamflow simulations of large watersheds with limited observation data. *Journal of Hydrology*, 590:125206, 2020.