

|  |  |
| --- | --- |
| Title of Project: | **Deep Learning of hydrological rating curves** |

|  |  |
| --- | --- |
| **Principal Investigator** | Gabriel Hernandez Ramirez |
| Student at Indiana University / Visiting Researcher |
| Bachelor of Science in Informatics |
| hernaga@iu.edu |

|  |  |
| --- | --- |
| **Research advisor** | Dr. Dipsikha Devi & Dr. Anupal Baruah |
| Post Doctoral Researcher, Surface Dynamics Modelling Lab |
| Department of Geography |
| University of Alabama |

**RESEARCH PROBLEM STATEMENT**

**Introduction**

Hydrological rating curves are essential tools in hydrology, providing a relationship between the water level and the flow rate of a river or stream at a particular location. These curves can be expressed mathematically with the power law which is , “where a, b and c (in this case a, x and b) are calibration parameters that are usually estimated by means of the non-linear least squares method (e.g. Petersen-Øverleir 2004). Equation (1) is widely used in river hydraulics and has some physical justifications (Chow 1959; Fenton 2001; Petersen-Øverleir 2005). More recently, Reitan & Petersen-Øverleir (2008) analyzed the use of power law (Equation (1)) segments to cope with stage–discharge relationships that change at certain flow stages. Hydrologic measurement standards require a periodic updating of the rating curve to account for changes that may occur in the river geometry. These updates produce annual rating curves that sometimes change considerably from one year to another.” (<https://iwaponline.com/hr/article/42/1/10/659/A-hydraulic-study-on-the-applicability-of-flood>, 11)

For this project, we are made available with 7000 data collection points in the United States for existing rating curves. I will work on a python program using ML and deep learning to be able to predict these rating curves at places where there isn’t a collection point.

**Objectives**

1. Develop an ML and deep learning-based model to predict hydrological rating curves.
2. Validate the model using existing data collection points.
3. Assess the model's performance and reliability in predicting rating curves at unmonitored locations.

**Research Challenges**

This project addresses several key challenges:

1. **Data Quality and Availability**: Ensuring the quality and completeness of the data from 7000 collection points.
2. **Model Complexity**: Developing a robust model that can handle the variability in hydrological data and predict accurate rating curves.
3. **Geographical Variability**: Accounting for geographical and environmental factors that influence the rating curves.
4. **Model Generalization**: Ensuring that the model generalizes well to new, unseen locations without data collection points.

**Research Question**

How can we develop an accurate and reliable model to predict hydrological rating curves for locations without data collection points using ML and deep learning techniques?

**Research Approach**

**To address the research question, the following approach will be taken:**

1. Data Collection and Preprocessing: Gather and preprocess data from 7000 existing collection points. This includes cleaning the data, handling missing values, and normalizing the data for model training.
2. Feature Engineering: Identify and engineer relevant parts of the data that influence the rating curves, such as river geometry, flow rate, and water level.
3. Model Development: Develop and train various ML and deep learning models, such as linear regression, decision trees, random forests, and neural networks, to predict the rating curves.
4. Model Validation: Validate the models using a subset of the data and evaluate their performance using metrics like mean squared error (MSE) and R-squared. Use some other metrics that are widely used in hydrology, such as Nash Sucliffe efficiency and Kling Gupta efficiency (Recommended by Andres Ramirez)
5. Model Testing and Deployment: Test the models on unseen locations and refine them based on performance. Deploy the best-performing model for practical use.

**Anticipated Outcomes:**

Through this approach, we expect to:

1. Develop a reliable model capable of predicting hydrological rating curves for locations without data collection points.
2. Enhance our understanding of the factors influencing rating curves and their geographical variability.
3. Provide a valuable tool for hydrologists and water resource managers to estimate flow rates and manage water resources more effectively.

**Be more precise on what the problem is.**

**Issue of having too much data… Overfitting and underfitting.**

**Think of cross validation techniques.**

**(Been finding several different formulas, are these all equivalent…? E.g. Page 11 in https://iwaponline.com/hr/article/42/1/10/659/A-hydraulic-study-on-the-applicability-of-flood)**

**(Talk about formula, what each variable stands for and how it looks on a graph)**

**Apparently, river sediment rating curves use the same power law. The Effects of Water and Sediment Discharge on Rating Parameters (**[**https://onlinelibrary-wiley-com.proxyiub.uits.iu.edu/doi/full/10.1002/hyp.10198**](https://onlinelibrary-wiley-com.proxyiub.uits.iu.edu/doi/full/10.1002/hyp.10198)**)**

**What type of data will be made available to me? -- I will need some sort of data about the places without a rating curve, whether that’s the water level, the flow rate or some sort of coordinates? – James thinks I’ll use the Digital Elevation Model and Digital terrain model to predict the rating curve where data isn’t collected.**

-Sounds like regression

-Divide data between training, validation and testing

- Overfeeding is when

-Hyperparameters are the ones that affect the whole model

-With the testing and training, you adjust hyperparamenters

-Once you optimize hyperparameters, you test the model in a new data set that hasn’t been looked.

- grid search python – sirve para optimizacion automatica de hiperparametros

- Library – Auto ML

-LSTM depending on if dates, fully connected networks

**- In deep networks, privilege profundity over complexity**

-Deep learning tiene muchas capas, usualmente 3 o más

- 10 epocas, alimentar un modelo 10 veces con los mismos datos.

-Early stopping to find the right epoch